

» Estás en: [Inicio](#) [Tutoriales](#) [Instalación de un clúster Hadoop con Cloudera-Manager](#)

Juan Antonio Cantarero

Ingeniero Informático y Jefe de Equipo en Proyectos para Telcos.

Puedes saber más sobre mí en LinkedIn: es.linkedin.com/in/juancantarero/[Ver todos los tutoriales del autor](#)
 Tu foto impresa en una
 original pieza con carcasa
 personalizable
por sólo
9,95€

Fecha de publicación del tutorial: 2015-02-25

Tutorial visitado 2 veces [Descargar en PDF](#)

Instalación de un clúster Hadoop con Cloudera-Manager

0. Índice de contenidos.

- 1. Prerrequisitos.
- 2. Introducción y objetivos.
- 3. Requisitos necesarios
- 4. Preparación del entorno
- 5. Instalación de Cloudera-Manager
- 6. Cuando todo falla.
- 7. Sigüientes pasos.

1. Prerrequisitos.

Si quieres aprovechar bien el contenido de este tutorial, deberás tener:

- conocimientos de la arquitectura de Hadoop. Nivel medio.
- conocimientos de administración de Linux. Nivel medio.
- conocimientos de administración de Redes/Seguridad. Nivel básico.

2. Introducción y objetivos.

El objetivo de este tutorial es aprender a instalar, configurar y monitorizar un cluster Hadoop en modo distribuido, mediante el framework *Cloudera-Manager Enterprise*, todo ello usando máquinas virtuales (mediante Vmware). Espero que os sea útil.

Cloudera-Manager es una plataforma de administración de Cloudera open source, para la gestión de Clústers Hadoop. Este tipo de frameworks facilitan la gestión manual que supone la administración de un clúster, ya que se trata de un trabajo complicado y bastante propenso a errores. Pensar en todos los pasos que hay que seguir en cada nodo: instalación del paquete de hadoop, configuración de variables de entorno, definición de archivos de configuración, definir permisos y reglas de seguridad, levantar demonios, formateo del sistema HDFS, etc.



Logo de Cloudera-Manager

Cloudera-Manager no es la única opción en el mercado como hemos visto en otros tutoriales (Instalación de un entorno Hadoop con Ambari), pero es la solución líder actualmente por varios motivos como veremos más adelante.

Definiremos un clúster distribuido mediante uso de máquinas virtuales, lo que nos permitirá "jugar" con cloudera-manager. Hay otras opciones a la hora de definir un clúster: por ejemplo podemos interconectar varios PCs, o bien podemos trabajar con servicios en la nube (EC2 de Amazon, Azure de Microsoft, RackSpace, etc.). La opción que veremos en este tutorial es la más asequible en muchos aspectos.

3. Requisitos necesarios.

Para seguir los pasos en este tutorial necesitaremos:

Catálogo de servicios Autentia



Síguenos a través de:



Últimas Noticias

» 2015: ¡Volvemos a la oficina!

» [Curso JBoss de Red Hat](#)

» Si eres el responsable o líder técnico, considérate desafortunado. No puedes culpar a nadie por ser gris

» Portales, gestores de contenidos documentales y desarrollos a medida

» [Comentando el libro Start-up Nation, La historia del milagro económico de Israel, de Dan Senor & Salu Singer](#)

[Histórico de noticias](#)

Últimos Tutoriales

» [Unicode](#)

» [Crea interfaces web amigables con Twitter Bootstrap](#)

» [Experimenta con tu código en Eclipse utilizando Scrapbooks](#)

» [Curso de WatchKit ¡ahora sólo 9 dólares!](#)

» [Cómo implementar una nube de etiquetas con D3.js](#)

- VMware Workstation versión 11 para Windows (cualquiera que pueda realizar Snapshots nos servirá). Yo he usado la versión 11 de evaluación de 30 días.
- Imagen de un Sistema Operativo Linux en VMware: lo usaremos como nodo inicial donde correrá el instalador de cloudera-manager. Yo he usado una máquina virtual Linux CentOS v6 (64 bits, requerido por Hadoop) y 8Gb RAM. Os recomiendo la distribución CentOS ya que es la más popular dentro de la comunidad Hadoop. Mientras mayor capacidad de recursos hardware tenga tu equipo, más nodos podrás incluir en tu clúster. Usar una distribución limpia de Hadoop, sin ninguna instalación activa o previa.

Cloudera manager acepta las siguientes distribuciones en 64 bits:

- Red Hat Enterprise Linux 5 (Update 7 or later recommended)
- Red Hat Enterprise Linux 6 (Update 4 or later recommended)
- Oracle Enterprise Linux 5 (Update 6 or later recommended)
- Oracle Enterprise Linux 6 (Update 4 or later recommended)
- CentOS 5 (Update 7 or later recommended)
- **CentOS 6 (Update 4 or later recommended)**
- SUSE Linux Enterprise Server 11 (Service Pack 2 or later recommended)
- Ubuntu 10.04 LTS (Only supports CDH 4.x)
- Ubuntu 12.04 LTS
- Ubuntu 14.04 LTS
- Debian 6.0 (Only supports CDH 4.x)
- Debian 7.0

Básicamente esto es todo lo que necesitas. En mi caso, al disponer de 8GB de RAM, he decidido definir la siguiente arquitectura:

1. Servidor primario para la instalación de cloudera-manager con: 3,5GB RAM+2 cores. Podría haber dedicado menos, pero prefiero trabajar cómodamente con el escritorio X11, por lo que requiere un poco más de RAM.
2. Cuatro (4) Nodos secundarios con los demonios Hadoop, cada uno de ellos con: 1 GB RAM+1 core+1 GB disco. Linux no requiere mucho más. Bien es cierto que Hadoop requiere más RAM, pero ya veremos que esto se puede resolver.

En total dedicaré el 90% de mi RAM disponible (3,5+4) al clúster en VMware.

4. Preparación del entorno.

Partimos de una imagen en Wmware tal y como se ha indicado anteriormente. Antes de proceder con el clonado de esta imagen, debemos realizar una serie de configuraciones en la misma de forma que ahorremos trabajo. Para ello:

1. Dentro de Linux, debemos desactivar la seguridad de SELinux (Security-Enhanced Linux) ya que de lo contrario no podremos continuar. Para ello editaremos el siguiente fichero:

```
1 | [user@ ~]$ sudo vi /etc/selinux/config
```

...cambiando la propiedad a "disabled":

```
1 | SELINUX=disabled
```

Comprobamos el nuevo estado mediante:

```
1 | [user@ ~]$ sestatus
2 | SELinux status: disabled
```

NOTA: estos comandos deben lanzarse mediante el usuario 'root' o en su defecto mediante un usuario en la lista de 'sudoers', tal y como yo estoy haciendo.

2. Durante la instalación, necesitamos tener abiertos ciertos puertos por lo que desactivaremos el firewall de CentOS mediante:

```
1 | [user@ ~]$ sudo chkconfig iptables off
2 | [user@ ~]$ sudo chkconfig ip6tables off
3 | [user@ ~]$ sudo /etc/init.d/iptables stop
```

3. Registramos las IPs de cada nodo en el fichero "/etc/hosts" por comodidad. Por ahora sólo conocemos la IP privada de la única máquina virtual que tenemos, pero podemos inventarnos el resto. Más adelante se corregirá. La IP privada se obtiene mediante (suponiendo que sólo tenemos una interfaz de red):

```
1 | [user@ ~]$ ifconfig -a
```

```
eth2      Link encap:Ethernet  HWaddr 00:0C:29:5C:AF:74
          inet addr:192.168.1.100  Bcast:192.168.35.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:365899 errors:0 dropped:0 overruns:0 frame:0
          TX packets:205707 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:528400737 (503.9 MiB)  TX bytes:28069743 (26.7 MiB)
          Interrupt:18 Base address:0x1424

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:128829 errors:0 dropped:0 overruns:0 frame:0
          TX packets:128829 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:66033622 (62.9 MiB)  TX bytes:66033622 (62.9 MiB)
```

```
1 | 192.168.1.100    cm
2 | 192.168.1.101    nodo1.hdp.hadoop
3 | 192.168.1.102    nodo2.hdp.hadoop
4 | 192.168.1.103    nodo3.hdp.hadoop
5 | 192.168.1.104    nodo4.hdp.hadoop
```

4. Los siguientes pasos nos permitirán acceder a cada nodo sin password, lo cual es recomendable durante la instalación de cloudera-manager. Para ello, generamos una clave pública sin password:

```

1 [user@ ~]$ ssh-keygen -t rsa
2
3 Generating public/private rsa key pair.
4 Enter file in which to save the key (/root/.ssh/id_rsa):
5 Enter passphrase (empty for no passphrase): ß No introducir clave
6 Enter same passphrase again: ß No introducir clave
7 Your identification has been saved in /root/.ssh/id_rsa.
8 Your public key has been saved in /root/.ssh/id_rsa.pub.
9 The key fingerprint is:
10 b5:cb:81:e9:c3:2f:ac:72:98:e0:54:71:23:30:e1:f9 root@ip-*****.eu-west-1.compute
11 The key's randomart image is:
12 +--[ RSA 2048]-----+

```

Esto nos genera el archivo “/root/.ssh/id_rsa.pub” con la clave pública. También debéis generar un fichero de clave privada (.pem) que será requerido posteriormente. Ambos ficheros (“.pub” y “.pem”) se añadirán a la lista de claves autorizadas de la siguiente forma:

```

1 [user@ ~]$ ssh-keygen -y -f mi_par_claves_AWS.pem >> .ssh/authorized_keys
2 [user@ ~]$ cat .ssh/id_rsa.pub >> .ssh/authorized_keys

```

Conviene reiniciar la máquina virtual en este punto, mediante “reboot”.

5. La distribución Linux de la que yo parto inicia el entorno gráfico X11. En mi caso haré que en el arranque se inicie en modo texto para que no consuma recursos innecesarios. Para ello en la última línea del fichero “/etc/inittab” reemplazamos el código 5 por 3, de forma que nos quede:

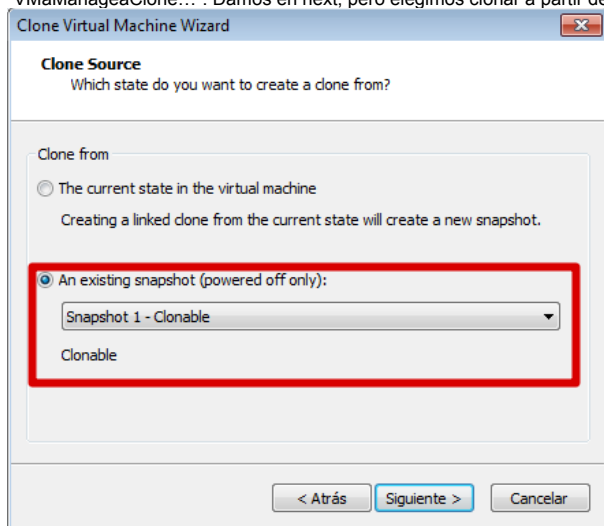
```

1 id:3:initdefault:

```

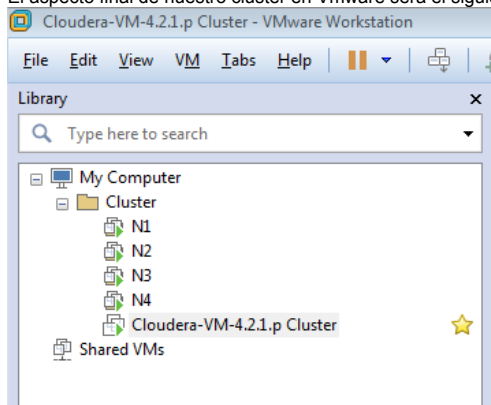
En mi caso, además tengo que adaptar el teclado modificando el fichero “/etc/sysconfig/keyboard”.

6. A continuación procedemos con el clonado de máquinas, para obtener los 4 nodos. Para ello apagamos la máquina virtual (Power Off) y la seleccionamos en Vmware para acceder al menú “VMàSnapshotàTake Snapshot...”. Esto sólo hay que hacerlo una vez. Posteriormente procedemos al clonado. Nuevamente sobre nuestra máquina virtual en Vmware, accedemos al menú “VMàManageàClone...”. Damos en next, pero elegimos clonar a partir de una snapshot:



Para el resto de ventanas elegiremos las opciones por defecto, salvo en el nombre de la máquina clonada que asignaremos nombres descriptivos como: nodo1, ..., nodoN.

El aspecto final de nuestro cluster en Vmware será el siguiente (podéis crear una carpeta para mayor claridad):



7. Antes de arrancar cada máquina, procedemos a configurar algunos parámetros en Vmware tal y como se ha indicado anteriormente:
 - a. Memoria: 1GB RAM.
 - b. CPUs: 1 core.
8. Procedemos a arrancar los nodos y en cada uno de ellos realizamos las siguientes acciones; cambiamos el nombre del host para que coincida con el configurado en el fichero de hosts:

```

1 [user@ ~]$ sudo hostname nodoX

```

Lo hacemos permanente editando la propiedad HOSTNAME en el fichero:

```

1 [user@ ~]$ sudo vi /etc/sysconfig/network

```

Actualizamos la IP de nuestro nodo en el fichero “/etc/hosts” tal y como hemos hecho antes.

5. Instalación de Cloudera-Manager.

1. Una vez preparado el entorno, procedemos a descargarnos el instalador de cloudera-manager mediante:

```

1 [user@ ~]$ wget http://archive.cloudera.com/cm5/installer/latest/cloudera-manager?.
2
3 --2015-- http://archive.cloudera.com/cm5/installer/latest/cloudera-manager-install
4 Resolving archive.cloudera.com...
5 Connecting to archive.cloudera.com|:80... connected.
6 HTTP request sent, awaiting response... 200 OK
7 Length: 514295 (502K) [application/octet-stream]
8 Saving to: "cloudera-manager-installer.bin"
9 100%[=====]
10 2015-01-30 12:10:43 (595 KB/s) - "cloudera-manager-installer.bin" saved [514295/514

```

2. Concederemos permisos de ejecución al paquete y lo ejecutaremos:

```

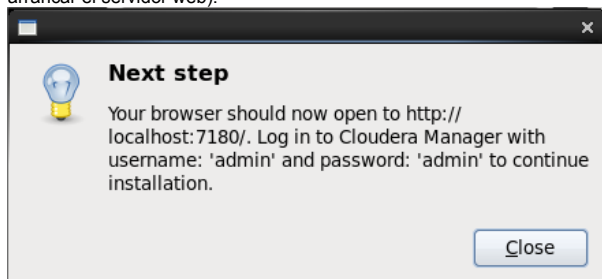
1 [user@ ~]$ chmod u+x cloudera-manager-installer.bin
2 [user@ ~]$ sudo ./cloudera-manager-installer.bin

```

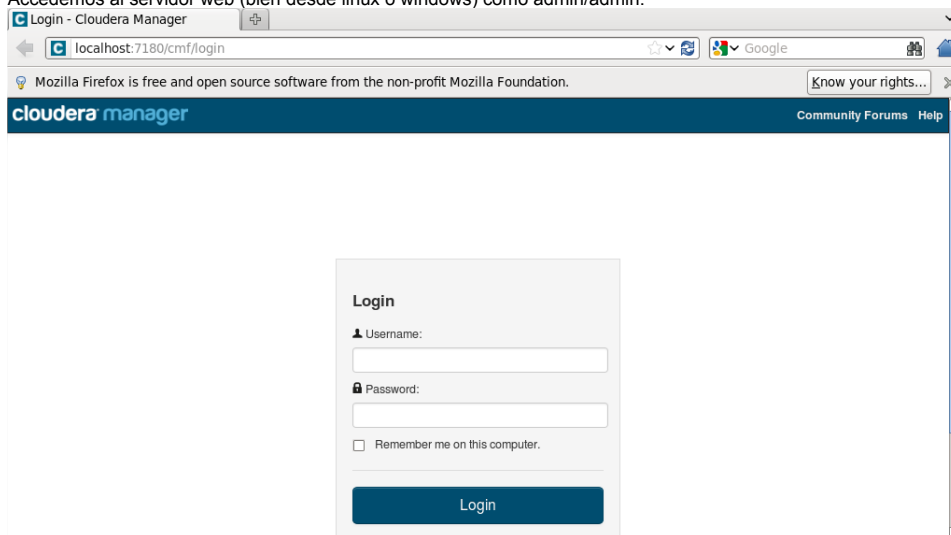
Nos aparecerán una serie de ventanas en las que iremos avanzando con las opciones por defecto (no es necesario instalar jdk 1.7):



Al finalizar nos saldrá el siguiente popup indicando que ya podemos acceder al servidor web (tardará unos minutos en arrancar el servidor web):



3. Accedemos al servidor web (bien desde linux o windows) como admin/admin:



Seleccionamos la versión Enterprise Data Hub:

cloudera managerSupport · admin

Welcome to Cloudera Manager. Which edition do you want to deploy?

Upgrading to **Cloudera Enterprise Data Hub Edition** provides important features that help you manage and monitor your Hadoop clusters in mission-critical environments.

Cloudera Express	Cloudera Enterprise Data Hub Edition Trial	Cloudera Enterprise
License	Free	60 Days
	After the trial period, the product will continue to function as Cloudera Express . Your cluster and your data will remain unaffected.	Annual Subscription
		Upload License

Cloudera Enterprise is available in three editions:

- Basic Edition
- Flex Edition
- Data Hub Edition

Indicamos la lista de nodos:

cloudera managerSupport · admin

Specify hosts for your CDH cluster installation.

Hosts should be specified using the same hostname (FQDN) that they will identify themselves with. Cloudera recommends including Cloudera Manager Server's host. This will also enable health monitoring for that host.

Hint: Search for hostnames and/or IP addresses using [patterns](#).

SSH Port: [Search](#)

Comprobamos que hay conectividad en todos los nodos:

Specify hosts for your CDH cluster installation.

Hosts should be specified using the same hostname (FQDN) that they will identify themselves with. Cloudera recommends including Cloudera Manager Server's host. This will also enable health monitoring for that host.

Hint: Search for hostnames and/or IP addresses using [patterns](#).

4 hosts scanned, 4 running SSH. [New Search](#)

☒	Expanded Query	Hostname (FQDN)	IP Address	Currently Managed	Result
☒	nodo1	nodo1.ejemplo.com	192.168.1.1	No	✓ Host ready: 13 ms response time.
☒	nodo2	nodo2.ejemplo.com	192.168.1.2	No	✓ Host ready: 13 ms response time.
☒	nodo3	nodo3.ejemplo.com	192.168.1.3	No	✓ Host ready: 11 ms response time.
☒	nodo4	nodo4.ejemplo.com	192.168.1.4	No	✓ Host ready: 14 ms response time.

Indicamos el usuario a usar (root si disponemos del mismo, o bien otro usuario con privilegios). También especificamos el fichero ".pem" con la clave privada ya generada anteriormente:

Cluster Installation

Provide SSH login credentials.

Root access to your hosts is required to install the Cloudera packages. This installer will connect to your hosts via SSH and log in either directly as root or as another user with password-less sudo/pbrun privileges to become root.

Login To All Hosts As: ☐ root ☒ Another user (with password-less sudo/pbrun to root)

You may connect via password or public-key authentication for the user selected above.

Authentication Method: ☐ All hosts accept same password ☒ All hosts accept same private key

Private Key File: [Browse...](#)

Enter Passphrase:

Confirm Passphrase:

SSH Port:

A continuación comenzará la instalación de cloudera-manager-agent en cada nodo (el tiempo de este paso depende de la velocidad de tu red):

Cluster Installation

Installation completed successfully.

4 of 4 host(s) completed successfully.

Hostname	IP Address	Progress	Status	
nodo1.ejemplo.com	192.168.65.14		✓ Installation completed successfully.	Details
nodo2.ejemplo.com	192.168.65.14		✓ Installation completed successfully.	Details
nodo3.ejemplo.com	192.168.65.14		✓ Installation completed successfully.	Details
nodo4.ejemplo.com	192.168.65.14		✓ Installation completed successfully.	Details

Una vez completada la instalación, avanzaremos por las siguientes ventanas con los valores por defecto:

Cluster Installation

Detecting CDH versions on all hosts

All hosts have the same CDH version.

- The following host(s) are running CDH 5.3.1: nodo[1-4].ejemplo.com

Cluster Setup

Choose the CDH 5 services that you want to install on your cluster.

Choose a combination of services to install.

☒ **Core Hadoop**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, and Sqoop

☐ **Core with HBase**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and HBase

☐ **Core with Impala**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and Impala

☐ **Core with Search**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and Solr

☐ **Core with Spark**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, and Spark

☐ **All Services**
HDFS, YARN (MapReduce 2 Included), ZooKeeper, Oozie, Hive, Hue, Sqoop, HBase, Impala, Solr, Spark, and Key-Value Store Indexer

☐ **Custom Services**
Choose your own services. Services required by chosen services will automatically be included. Flume can be added after your initial cluster has been set up.

This wizard will also install the **Cloudera Management Service**. These are a set of components that enable monitoring, reporting, events, and alerts; these components require databases to store information, which will be configured on the next page.

☒ **Include Cloudera Navigator**

- Please ensure that you have the appropriate license for **Cloudera Navigator** or contact Cloudera for assistance.
- The default audit event filter discards events generated by the internal Cloudera and Hadoop users (cloudera-scm, hdfs, hbase, hive, mapred, solr, and dr.who) and that affect files in the /tmp directory.

Customize Role Assignments

You can customize the role assignments for your new cluster here, but if assignments are made incorrectly, such as assigning too many roles to a single host, this can impact the performance of your services. Cloudera does not recommend altering assignments unless you have specific requirements, such as having pre-selected a specific host for a specific role.

You can also view the role assignments by host. [View By Host](#)

HDFS

1NN

NameNode × 1 New

Same As DataNode

2NN

SecondaryNameNode × 1 New

Same As DataNode

B

Balancer × 1 New

Same As DataNode

HDFS

HttpFS

Select hosts

NFS

NFS Gateway

Select hosts

DN

DataNode × 1 New

nodo1.ejemplo.com

Hive

G

Gateway × 1 New

Same As DataNode

HMS

Hive Metastore Server × 1 New

Same As DataNode

HC2

WebHCat Server

Select hosts

HS2

HiveServer2 × 1 New

Same As DataNode

Hue

Configure and test database connections. Create the databases first according to the **Installing and Configuring an External Database** section of the [Installation Guide](#) ¹.

☒ Use Custom Databases

☐ Use Embedded Database

Hive

✓ Skipped. Cloudera Manager will create this database in a later step.

Database Host Name: *

Database Type:

Database Name: *

Username: *

Password:

cm.ejemplo.com:7432

PostgreSQL

hive

hive

Reports Manager

✓ Successful

Currently assigned to run on **nodof1.ejemplo.com**.

Database Host Name: *

Database Type:

Database Name: *

Username: *

Password:

cm.ejemplo.com:7432

PostgreSQL

rman

rman

Navigator Audit Server

✓ Successful

Currently assigned to run on **nodof1.ejemplo.com**.

Database Host Name: *

Database Type:

Database Name: *

Username: *

Password:

cm.ejemplo.com:7432

PostgreSQL

nav

nav

Navigator Metadata Server

✓ Successful

Currently assigned to run on **nodof1.ejemplo.com**.

Database Host Name: *

Database Type:

Database Name: *

Username: *

Password:

cm.ejemplo.com:7432

PostgreSQL

navms

navms

☐ Show Password

Test Connection

Review Changes

DataNode Data Directory

dfs.data.dir, dfs.datanode.data.dir

DataNode Default Group

/dfs/dn

+ -

Comma-delimited list of directories on the local file system where the DataNode stores HDFS block data. Typical values are /data/N/dfs/dn for N = 1, 2, 3... These directories should be mounted using the noatime option and the disks should be configured using JBOD. RAID is not recommended.

DataNode Failed Volumes Tolerated

dfs.datanode.failed.volumes.tolerated

DataNode Default Group

0

The number of volumes that are allowed to fail before a DataNode stops offering service. By default, any volume failure will cause a DataNode to shutdown.

NameNode Data Directories

dfs.name.dir, dfs.namenode.name.dir

NameNode Default Group

/dfs/nn

+ -

Determines where on the local file system the NameNode should store the name table (fsimage). For redundancy, enter a comma-delimited list of directories to replicate the name table in all of the directories. Typical values are /data/N/dfs/nn where N=1..3.

HDFS Checkpoint Directory

fs.checkpoint.dir, dfs.namenode.checkpoint.dir

SecondaryNameNode Default Group

/dfs/snn

+ -

Determines where on the local file system the DFS SecondaryNameNode should store the temporary images to merge. For redundancy, enter a comma-delimited list of directories to replicate the image in all of the directories. Typical values are /data/N/dfs/snn for N = 1, 2, 3...

Review Configuration for Single User Mode

The configured single user for the cluster must have permissions to create or write to the directories below. Typically, users create a top-level directory for each data drive (for example, /data/1/cloudera, /data/2/cloudera) owned by the configured user and prefix the paths below with those top-level directories.

For example, in the above case, change the path values below from /data/dfs/... to /data/1/cloudera/dfs/... and /data/2/cloudera/dfs/..., change the path values below from /var/... to /data/1/cloudera/var/...

DataNode Data Directory

dfs.data.dir, dfs.datanode.data.dir

DataNode Default Group

/dfs/dn

+ -

Comma-delimited list of directories on the local file system where the DataNode stores HDFS block data. Typical values are /data/N/dfs/dn for N = 1, 2, 3... These directories should be mounted using the noatime option and the disks should be configured using JBOD. RAID is not recommended.

NameNode Data Directories

dfs.name.dir, dfs.namenode.name.dir

NameNode Default Group

/dfs/nn

+ -

Determines where on the local file system the NameNode should store the name table (fsimage). For redundancy, enter a comma-delimited list of directories to replicate the name table in all of the directories. Typical values are /data/N/dfs/nn where N=1..3.

HDFS Checkpoint Directory

fs.checkpoint.dir, dfs.namenode.checkpoint.dir

SecondaryNameNode Default Group

/dfs/snn

+ -

Determines where on the local file system the DFS SecondaryNameNode should store the temporary images to merge. For redundancy, enter a comma-delimited list of directories to replicate the image in all of the directories. Typical values are /data/N/dfs/snn for N = 1, 2, 3...

A continuación se iniciarán cada uno de los demonios en cada nodo:

Progress

Command	Context	Status	Started at	Ended at
First Run		In Progress	Feb 10, 2015 6:15:54 PM CET	

Command Progress

Completed 1 of 22 steps.

✓ Initializing ZooKeeper Service

Completed 1 steps successfully.

Starting ZooKeeper Service

[Details](#)

Checking if the name directories of the NameNode are empty. Formatting HDFS only if empty.

Starting HDFS Service

Creating HDFS /tmp directory

Creating MR2 job history directory

Creating NodeManager remote application log directory

...y se nos mostrará la ventana principal:

Ya tenemos listo nuestro clúster hadoop gestionado con Cloudera Manager. En la pestaña de "Inicio" podemos ver el estado general del cluster y los problemas si los hay. Conviene revisar cada uno de los Hosts para comprobar que todos los demonios están levantados:

6. Cuando todo falla.

Uno de los inconvenientes del proceso de instalación en este tipo de herramientas es que lanzan scripts batch (python, perl, etc.) "transparentes" al usuario para avanzar con la instalación. Las acciones que estos scripts ejecutan, dependen de muchas variables del entorno para que terminen satisfactoriamente (permisos y credenciales, versiones de software, dependencias con paquetes rpm, seguridad, puertos y reglas del proxy, requisitos hardware, etc.).

Lo más probable es que tengáis que repetir el proceso de instalación varias veces hasta que tengáis todos los requisitos hard/soft bien configurados. La gran mayoría ya se han tratado en este tutorial pero siempre puede quedar alguno por comentar, lo cual depende de la configuración de vuestro SO. Vamos a comentar a continuación cómo salir airoso ante un error durante la instalación de cloudera manager.

Lo primero es saber el estado de nuestra instalación. ¿Sigue en curso? ¿Está detenida por algún error?. Para ello podemos consultar los procesos en busca del que nos preocupa:

```
1 [user@ ~]$ sudo ps -aef | grep yum ?
2 root      7280   7136   0 12:18 pts/0    00:00:00 sh -c DEBIAN_FRONTEND=noninteractive yum
3 root      7281   7280   9 12:18 pts/0    00:00:03 /usr/bin/python /usr/bin/yum -y install c
```

Si vemos que están idle y no progresan, mala pinta. Aunque lo más elegante es seguir los logs de la instalación mediante:

```
1 [user@ ~]$ cd /var/log/cloudera-manager-installer/ ?
2 [user@ ~]$ tail -f install-cloudera-manager-server.log
```

En caso de finalizar la instalación ¿Cómo sé el estado de mi servidor de cloudera manager? Con el siguiente comando es fácil:

```
1 [user@ ~]$ service cloudera-scm-server status ?
2 cloudera-scm-server (pid 2627) is running...
```

Aunque personalmente me fio más del ps. Siempre habrá 3 procesos corriendo si todo va bien: el servidor de la BBDD Postgres, el servidor de cloudera, y el job java asociado:

```
1 [user@ ~]$ ps -aef | grep scm ?
2 485      10938     1 0 16:04 ?        00:00:25 /usr/bin/postgres -D /var/lib/cloudera-sc
3 root      10978     1 0 16:05 pts/0    00:00:00 su cloudera-scm -s /bin/bash -c nohup /us
4 485      10980 10978 11 16:05 ?        00:22:28 /usr/java/jdk1.7.0_67-cloudera/bin/java -
```

Si estás en un callejón sin salida en la instalación y queréis volver a empezar desde cero, tenéis varias opciones. La elegante pero que no suele funcionar, es ejecutar el desinstalador:

```
1 [user@ ~]$ sudo /usr/share/cmf/uninstall-cloudera-manager.sh ?
```

La otra opción es limpiar nuestro equipo de la instalación de cloudera. Para ello:

1. Parar todos los servidores de cloudera manager (si no funciona, matar los procesos con "kill"):

```
1 [user@ ~]$ sudo service cloudera-scm-server start ?
2 Stopping cloudera-scm-server: [ OK ]
3 [user@ ~]$ sudo service cloudera-scm-server-db stop
4 DB initialization done.
5 waiting for server to stop.... done
6 server stopped
```

2. Eliminar todo rastro de la instalación en el repositorio yum (si no funciona, forzar el borrado con rpm: "rpm -e --noscripts"):

```
1 [user@ ~]$ cd /etc/yum.repos.d ?
2 [user@ ~]$ sudo yum remove 'cloudera-manager-*'
3 [user@ ~]$ sudo rm -Rf /usr/share/cmf /var/lib/cloudera* /var/cache/yum/cloudera*
```

3. Limpiar la cache de yum:

```
1 [user@ ~]$ sudo yum clean all ?
```

En ocasiones, al volver a iniciar una nueva instalación, puede aparecer el mensaje: "cloudera-scm-server dead but pid file exists". Para resolverlo debemos detener todos los servidores de cloudera manager y eliminar el siguiente fichero:

```
1 [user@ ~]$ sudo rm /var/run/cloudera-scm-server.pid ?
```

El mismo procedimiento debemos seguir cuando el proceso de instalación se quede bloqueado en el paso "cloudera manager acquiring installation lock". Debemos borrar el fichero de bloqueo en cada uno de los nodos:

```
1 [user@ ~]$ sudo rm /tmp/.scm_prepare_node.lock ?
```

Durante el proceso de instalación, tener en cuenta que:

- Podemos "abandonarlo" temporalmente, cerrando el explorador sin problemas. Cuando deseemos continuar, basta con volverse a conectarse al puerto 7180. La instalación seguirá en background.
- No podemos cambiar ninguna IP (en caso de trabajar con IPs dinámicas). De ser así, la instalación fallará y la solución pasa por cambiar las IPs manualmente en la BBDD postgres de cloudera manager. Nada recomendado.
- La navegación dentro del proceso de instalación no está muy fina, y a veces se nos muestra la opción de pulsar en el botón "Back" para volver al paso anterior, cuando realmente no es posible. Si estáis en este punto, sólo os queda volver a comenzar ("Abort instalation").
- El botón "Abort instalation" cancela la instalación obligando a iniciarla desde el principio, pero no limpia el sistema de archivos temporales.
- La instalación hace uso de caché, de forma que sólo el primer intento es el que lleva tiempo. Los subsiguientes son más rápidos.

Un aspecto positivo a comentar de cloudera manager sobre otras soluciones similares, es que el proceso de instalación no se detiene si alguno de los demonios (maestro o esclavo) no arranca en algún nodo, siempre que no sea un demonio vital como el NameNode para HDFS. A posteriori podremos solucionar el problema y reintentar iniciar el demonio que daba problemas.

7. Siguiendo pasos.

Una vez terminado de instalar nuestro clúster, los siguientes pasos que recomiendo (ya fuera de este tutorial) serían lanzar un job hadoop y ver el resultado desde el cuadro de mando en el Host Monitor, así como estudiar las distintas métricas que nos proporciona cloudera manager. Lo más rápido para ello, es usar la librería de ejemplos que viene con la distribución descargada, y lanzar por ejemplo el estimador del número pi para 12 mappers y 3000 muestras:

```
1 [user@ ~]$ sudo -u hdfs hadoop jar /usr/lib/hadoop-mapreduce/hadoop-mapreduce-examples.jar ?
```

Otro ejercicio interesante que nos permite cloudera manager es jugar con los nodos: tirar un nodo, moverlo de ubicación, añadir nodos, y volver a lanzar nuestro job hadoop para ver los cambios introducidos en el cluster.

También por comodidad, conviene hacer las IPs de cada nodo (o máquina virtual) estáticas, para que no cambien entre reinicios de Vmware. Para ello:

1. Modificar el fichero de configuración de la interfaz de red, editando el archivo:

```
1 [user@ ~]$ sudo vi /etc/sysconfig/network-scripts/ifcfg-eth3
```

?

El nombre del fichero puede variar pero normalmente será "ethN". Debe corresponderse con el que sale al ejecutar:

```
1 [user@ ~]$ ifconfig -a
```

?

```
eth2: Link encap:Ethernet HWaddr 08:00:27:00:00:02
       inet addr:192.168.1.101 Bcast:192.168.255.255 Mask:255.255.255.0
       UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
       RX packets:126641 errors:0 dropped:0 overruns:0 frame:0
       TX packets:117882 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:1000
       RX bytes:27372574 (26.1 MiB) TX bytes:83449319 (79.5 MiB)
       Interrupt:18 Base address:0x1424
```

Debéis poner (el resto de parámetros los mantenéis):

```
1 BOOTPROTO=static
2 IPADDR=192.168.XXX.XXX
3 NETMASK=255.255.255.0
4 GATEWAY=192.168.YYY.YYY
5 ONBOOT=yes
```

?

La IP estática la asignáis vosotros (por ejemplo para los 4 nodos: ...101, ...102, ...103 y ...104) y el gateway será el de vuestro router. También debéis indicar el nombre del servidor que deseéis en el siguiente archivo:

```
1 [user@ ~]$ sudo vi /etc./resolv.conf
```

?

Debéis poner:

```
1 nameserver 192.168.XXX.XXX
```

?

Listo. Reiniciad la red con:

```
1 [user@ ~]$ sudo service network restart
```

?

Comprobad que tenéis la IP que habéis definido.

Espero que os sirva este tutorial para poder configuraros vuestro propio cluster.

Un saludo. Juan.

A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

Por favor, vota +1 o compártelo si te pareció interesante

Share |

 +1  0

Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

» **Regístrate** y accede a esta y otras ventajas «



Esta obra está licenciada bajo licencia [Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)