Installation Guide of a Cloudera cluster on Azure Cloude

*E*

Guía del usuario  
Manual Técnico

07 September 2019

Index

[1) Introduction 2](#_Toc14792623)

[2) Plataform requirements 3](#_Toc14792624)

[3) Descripció dels Discs utilitzats 4](#_Toc14792625)

[4) Rol Distribution 6](#_Toc14792626)

[5) Llista de Tasques a executar en Azure 7](#_Toc14792627)

[6) Infrastructure deployment on Azure 8](#_Toc14792628)

[a) Define Availability Set y FD domains 8](#_Toc14792629)

[b) Create VMs 8](#_Toc14792630)

[c) Create a Data Disk for the instances 10](#_Toc14792631)

[7) Systems setup 12](#_Toc14792632)

[a) Attaching a Disk to the instance 12](#_Toc14792633)

[b) DataNode 15](#_Toc14792634)

[c) Tune the OS 15](#_Toc14792635)

[d) Afegir el Repositori del Cloudera 16](#_Toc14792636)

[8) MySQL Database installation 16](#_Toc14792637)

[9) HA setup on the Database 19](#_Toc14792638)

[10) Install MySQL JDBC Driver 23](#_Toc14792639)

[11) Create DB Tables 23](#_Toc14792640)

[12) DataBase HUE i OOZIE 24](#_Toc14792641)

[13) Install Oracle JDK 24](#_Toc14792642)

[14) Instal.lar els paquets del Cloudera Manager 25](#_Toc14792643)

[15) Configurar el Cloudera Manager Server Database 25](#_Toc14792644)

[16) Install CDH on Cloudera Manager Server Database 26](#_Toc14792645)

[17) Configurar HDFS HA i Automatic Failover 33](#_Toc14792646)

[18) Setup YARN High Availability 36](#_Toc14792647)

[19) Connecting to Hive 37](#_Toc14792648)

[20) Private IP addressing and /etc/host file configuration 38](#_Toc14792649)

# Introduction

This document details the steps to follow to install a Hadoop cluster using the Cloudera software distribution. The cluster is deployed on top of Microsoft Azure platform.

The following documentation has been used.

Blog entry:

<https://blog.cloudera.com/blog/2016/02/how-to-install-cloudera-enterprise-on-microsoft-azure-part-1>

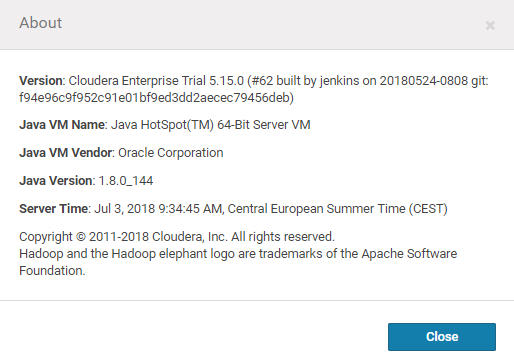
GitHub commands

<https://github.com/Azure/azure-quickstart-templates/tree/master/cloudera-on-centos>

Cloudera installation guide:

[https://www.cloudera.com/https://www.cloudera.com/documentation/enterprise/latest/topics/cm\_intro\_primer.htmldocumentation/enterprise/latest/topics/cm\_intro\_primer.html](https://www.cloudera.com/documentation/enterprise/latest/topics/cm_intro_primer.html)

“Installation Path B - Installation Using Cloudera Manager Parcels or Packages”



# Plataform requirements

The following list depict the requirements and the assumptions make during the installation:

* Hadoop cluster will be deployed without Kerberos on 4 nodes:
  + 2 Nodes as a Master Nodes
  + 2 Nodes as a Worker Nodes
* Cloudera Manager deployed without HA
* As recomended by Cloudera, an external database will be used, MySQL. This database will be deployed on HA
* This installation will be done using the Conventional mode (command run as root) instead of “Single User” mode
* Communication:
  + Outbound access to Internet des dels nodes a Internet
  + Inbound access to the platform only accesible from ClearPeaks public IPs. Security Groups will be created
* Network architecture.
  + All VMs will be deployed under the same private network. Therefore all VMs will have communication across their ports.
* Cloudera Manager version: CDH 5.15
* OS: CentOS 7.4
* Gold Image: Cloudera CentOS 7.4
* Cloudera manager will be used as software distribution. Parcel to distribute packages

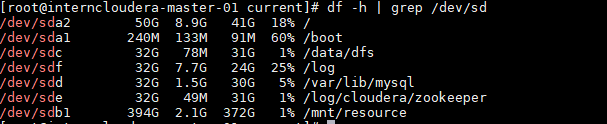
# Descripció dels Discs utilitzats

**VM A Master**

Resources:

* Disks:
  + 50 GB HDD - OS
  + 32 GB HDD - Log
  + 32 GB HDD - Zookeeper
  + 32 GB HDD - QJN
  + 32 GB HDD – MysqL

Disk consumption example:



*The OS root partition (where also the /var/log directory resides) is fairly small (10GB). This is perfect for an OS disk, but Cloudera also puts the parcels (an alternate form of distribution for Cloudera Hadoop) on /opt/cloudera and the logs into /var/logs. These take up quite a lot of space so a 1 GB disk is not enough. That’s why you should move the parcels and the log file to a different disk. (Reminder: normally the template takes care of this for you.)If you install Cloudera without moving these files to a different disk, you will see warning messages in Cloudera Manager that there not enough free disk space available.*

[*https://www.cloudera.com/documentation/other/reference-architecture/topics/ra\_bare\_metal\_deployment.html#concept\_m31\_nnl\_f2b*](https://www.cloudera.com/documentation/other/reference-architecture/topics/ra_bare_metal_deployment.html#concept_m31_nnl_f2b)

This guide shows how to deployed the disks:

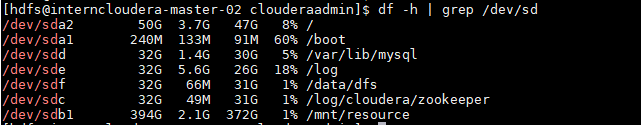
https://github.com/Azure/azure-quickstart-templates/blob/master/cloudera-on-centos/scripts/prepare-masternode-disks.sh

**VM B Master**

Disks:

* + 50 GB HDD - OS
  + 32 GB HDD - Log
  + 32 GB HDD - Zookeeper
  + 32 GB HDD - QJN
  + 32 GB HDD – MySQL

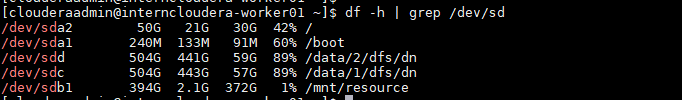
Disk consumption example:



**VM C Worker**

Disks:

* + 500 GB SSD - OS i logs
  + 1 TB HDD (2 x 0,5 GB)
* Disk consumption example:

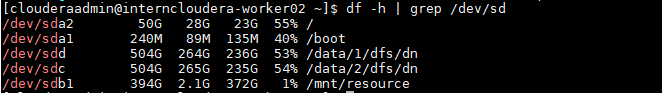


**VM D Worker**

Disks:

* + 500 GB SSD - OS i logs
  + 1 TB HDD (2 x 0,5 GB)

Disk consumption example:



**VM E Tools**

Resources:

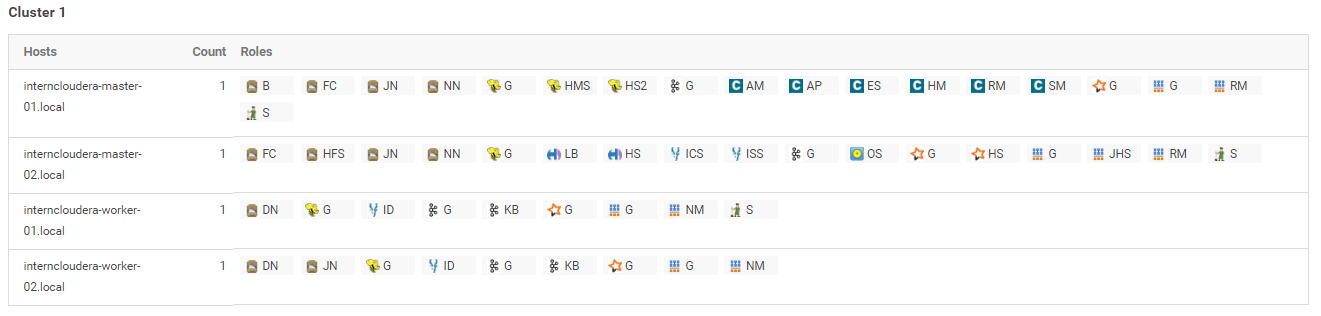
* Data Node
* Disks:

Remember Azure will create a temporary disk:

*Temporary disk The virtual machine is created with a temporary disk. This disk is stored in a physical drive of the host computer. It is not saved in Azure Storage and may be deleted during reboots and other events in the virtual machine life cycle. Use this disc only for temporary data, such as paging or swap files. For virtual machines with Linux, the temporary disk is / dev / sdb1 and is mounted on / mnt / resource or / mnt.*

# Role Distribution

Following image shows how rols are distributed on the different Nodes:



|  |  |
| --- | --- |
| Node | Rols |
| Node A - Master | Master Roles:  - NameNode  - Journal Node  - Failover Controller  - YARN Resource Manager  - Zookeeper  - MySQL DB - Master  - JobHistory Server  xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  Utility Roles:  - Clouder Manager  - Clouder Manager Management Service?  - Hive Metastore  - Hue  - Oozie |
| Node B - Master | Master Roles:  - NameNode  - Journal Node  - Failover Controller  - YARN Resource Manager  - Zookeeper  - MySQL DB - Slave |
| Node C - Worker | Worker Roles:  -Data Node  - Node Manager  - Impalad  Utility Roles:  - Journal Node  - Zookeeper  - Kafka Broker |
| Node D - Worker | Worker Roles:  -Data Node  - Node Manager  - Impalad  Utility Roles:  - Journal Node  - Zookeeper  - Kafka Broker |

# Llista de Tasques a executar en Azure

Segons descrit en el següent article, la llista de tasques a realitzar quan es vol desplegar un clúster de Hadoop en Azure són les següents:

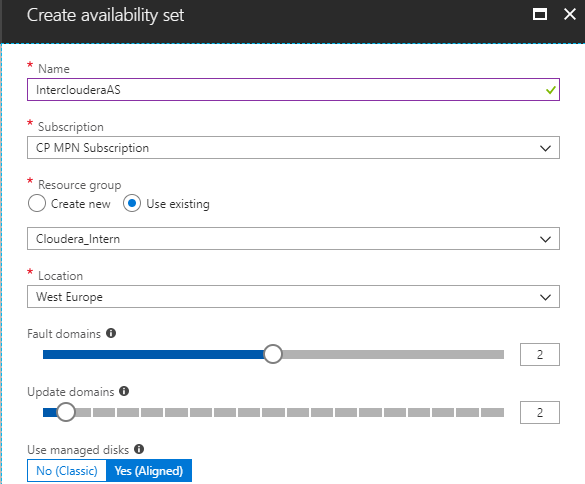
<https://blog.cloudera.com/blog/2016/02/how-to-install-cloudera-enterprise-on-microsoft-azure-part-1/>

* Create a Resource Group for all the components
  + Creem el RG: Cloudera\_Intern
* Create VNet and subnets
  + Only 1 Vnet. All VMs connect to the same Vnet
* Create availability sets. Place masters and workers in different availability sets
  + We create 1 availability set for Master nodes and 1 availability set for Worker nodes.
* Create security groups
* Create Masternode and Workernode instances using the Cloudera VM Image (CentOS image built and maintained by Cloudera). The template automatically uses Azure DS14 machines, which are the only machine types recommended and supported by Cloudera for Hadoop installations.
  + VMs will be create using the same certificate
* For each host a Premium Storage account is created
  + We use Standard Storage
* Add disks to the machines, format and mount the disks (10 data disks of 1 TB per node)
* Set up forward/reverse lookup between hosts using /etc/hosts file
  + We define the host files
* Tune Linux OS and network configurations like disable SELinux, disable IPtables, TCP tuning parameters, disable huge pages
* Set up time synchronization to an external server (NTPD)
  + Setup NTP
* Set up Cloudera Manager and the database used by the Cloudera Manager
  + Execute Cloudera Manager
* Set up Hadoop services using the Cloudera Python API

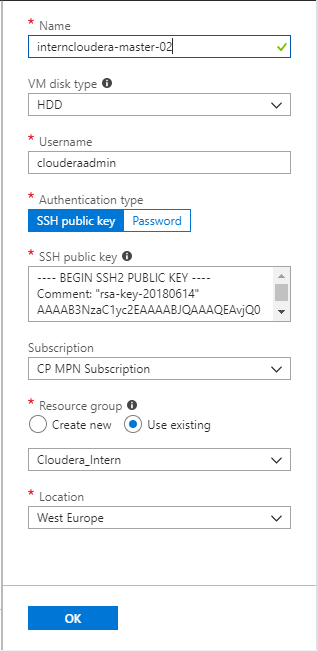
# Infrastructure deployment on Azure

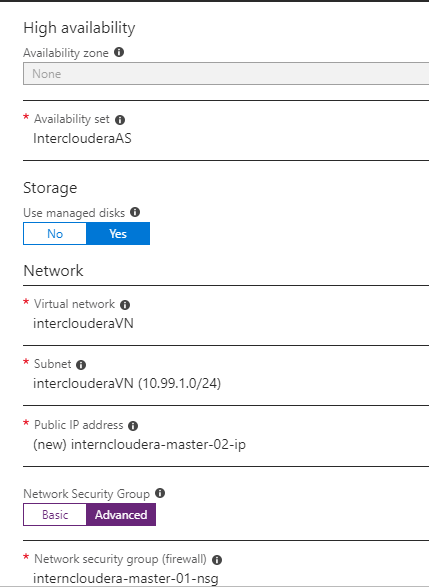
## Define Availability Set y FD domains

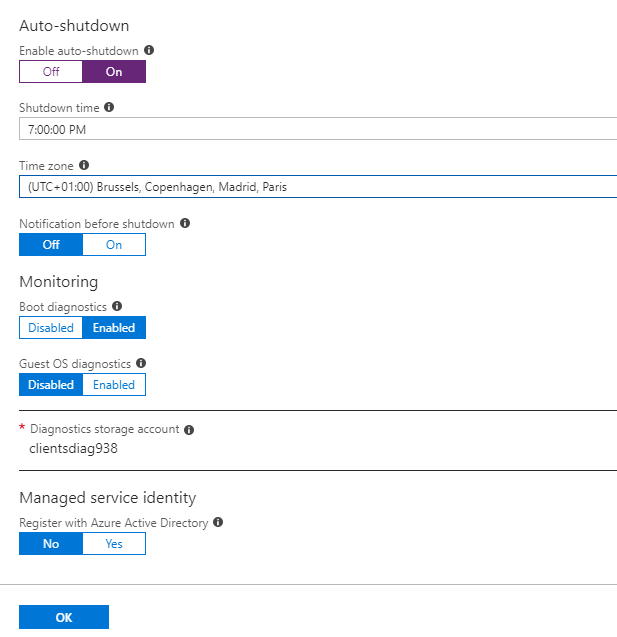
“Cluster Availability through Azure Availability Sets “ define AS and FD



## Create VMs

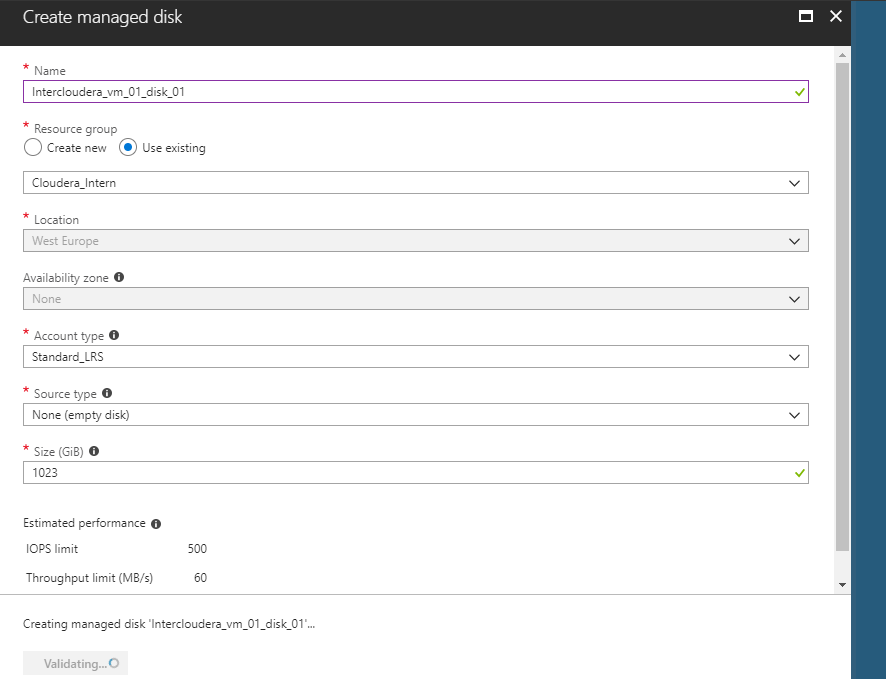


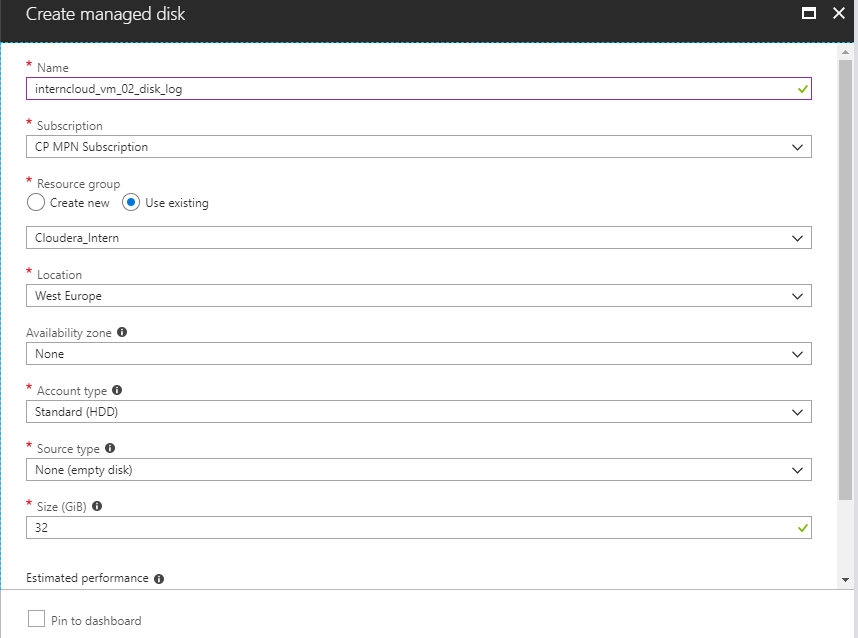




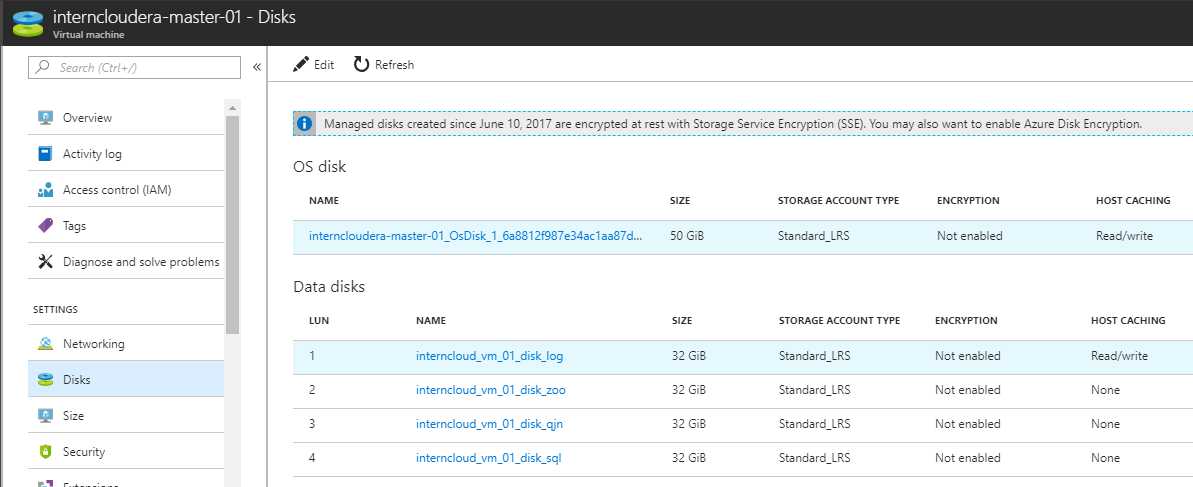
## Create a Data Disk for the instances

We create Standard Disk of 1 TB





Fem un attach dels disks a la VM:



# Systems setup

## Attaching a Disk to the instance

The following tables described the mapping between disks, mounting points and Azure identifiers

VM Master 01

|  |  |  |  |
| --- | --- | --- | --- |
| Azure LUN ID | Azure | Map LUN – DISK | UUID |
| 1 | /log | lun1 -> ../../../sde | 3f798317-1796-4779-8366-bc728cd45b0e |
| 2 | /log/cloudera/zookeeper | lun2 -> ../../../sdf | 1f59e7a0-5ca8-4365-8e26-6d8b097edaa0 |
| 3 | /data/dfs | lun3 -> ../../../sdd | 47711882-7429-4513-a839-5e345759eb68 |
| 4 | /var/lib/mysql | lun4 -> ../../../sdc | f31bbc2a-a320-447e-b32d-ef90b331c6a1 |

VM Master 02

|  |  |  |  |
| --- | --- | --- | --- |
| Azure LUN ID | Azure | Map LUN – DISK | UUID |
| 1 | /log | lun1 -> ../../../sdf | a5f967b2-a24e-4a7c-a6f7-c15d2208ce8b |
| 2 | /log/cloudera/zookeeper | lun2 -> ../../../sde | fcf37d89-dcbd-479c-8eed-4891b011c637 |
| 3 | /data/dfs | lun3 -> ../../../sdd | d0758f88-6178-4cfc-8041-2b20d30a5367 |
| 4 | /var/lib/mysql | lun4 -> ../../../sdc | 9a289862-e518-4087-a92c-c266f91c089a |

VM Woker 01

|  |  |  |  |
| --- | --- | --- | --- |
| Azure LUN ID | Azure | Map LUN – DISK | UUID |
| 1 | /mnt/resource/dfs/dn01vol01 | lun1 -> ../../../sdd | d3a031b2-5980-46c6-82e5-4c05301c4e08 |
| 2 | /mnt/resource/dfs/dn01vol02 | lun2 -> ../../../sdc | 5e2c3fd5-7966-4331-b171-b8a6605fa742 |

VM Woker 02

|  |  |  |  |
| --- | --- | --- | --- |
| Azure LUN ID | Azure | Map LUN – DISK | UUID |
| 1 | /mnt/resource/dfs/dn02vol01 | lun1 -> ../../../sdd | 12ae40ba-8fb3-4e55-8af0-6b961675994d |
| 2 | /mnt/resource/dfs/dn02vol02 | lun2 -> ../../../sdc | 0e0b33a9-059e-434e-9682-7df418356120 |

We follow the procedure to setup the disks:

https://docs.microsoft.com/en-us/azure/virtual-machines/linux/add-disk

|  |
| --- |
| dmesg | grep SCSI  [ 5527.508484] sd 5:0:0:1: [sdc] Attached SCSI disk  [clouderaadmin@interncloudera01 opt]$ lsblk  NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT  fd0 2:0 1 4K 0 disk  sda 8:0 0 50G 0 disk  ├─sda1 8:1 0 256M 0 part /boot  └─sda2 8:2 0 49.8G 0 part /  sdb 8:16 0 400G 0 disk  └─sdb1 8:17 0 400G 0 part /mnt/resource  sdc 8:32 0 32G 0 disk  sdd 8:48 0 32G 0 disk  sde 8:64 0 32G 0 disk  sdf 8:80 0 32G 0 disk  sr0 11:0 1 628K 0 rom  sudo mke2fs -F -t ext4 -b 4096 -E lazy\_itable\_init=1 -O sparse\_super,dir\_index,extent,has\_journal,uninit\_bg -m1 /dev/sdc  mkdir /log  mount -o noatime,barrier=1 -t ext4 /dev/sdc /log  [clouderaadmin@interncloudera01 dev]$ df -h  Filesystem Size Used Avail Use% Mounted on  /dev/sda2 50G 961M 49G 2% /  devtmpfs 14G 0 14G 0% /dev  tmpfs 14G 0 14G 0% /dev/shm  tmpfs 14G 8.4M 14G 1% /run  tmpfs 14G 0 14G 0% /sys/fs/cgroup  /dev/sda1 240M 89M 135M 40% /boot  /dev/sdb1 394G 2.1G 372G 1% /mnt/resource  tmpfs 2.8G 0 2.8G 0% /run/user/1000  /dev/sdc 32G 49M 31G 1% /log  [clouderaadmin@interncloudera01 dev]$ sudo lsblk -no UUID /dev/sdc  3e08a860-f4d2-4155-b06a-898efcad2083  [clouderaadmin@interncloudera01 dev]$ echo "UUID=3e08a860-f4d2-4155-b06a-898efcad2083 /log ext4 defaults,noatime 0 1" | sudo tee -a /etc/fstab  [clouderaadmin@interncloudera-master-01 log]$ lsblk  NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT  fd0 2:0 1 4K 0 disk  sda 8:0 0 50G 0 disk  ├─sda1 8:1 0 256M 0 part /boot  └─sda2 8:2 0 49.8G 0 part /  sdb 8:16 0 400G 0 disk  └─sdb1 8:17 0 400G 0 part /mnt/resource  sdc 8:32 0 32G 0 disk /data/dfs  sdd 8:48 0 32G 0 disk /var/lib/mysql  sde 8:64 0 32G 0 disk /log  sdf 8:80 0 32G 0 disk /log/cloudera/zookeeper |

On the /etc/fstab we configure the disks as: defaults,noatime. Recomenació de Cloudera

/etc/fstab File

|  |
| --- |
| UUID=0e15bce2-f88f-4d4c-a944-e8e0d283672e / xfs defaults 0 0  UUID=47906702-3436-4c20-98c0-c1328afbea34 /boot ext4 defaults 1 2  UUID=3f798317-1796-4779-8366-bc728cd45b0e /log ext4 defaults,noatime 0 1  UUID=f31bbc2a-a320-447e-b32d-ef90b331c6a1 /log/cloudera/zookeeper ext4 defaults,noatime 0 1  UUID=47711882-7429-4513-a839-5e345759eb68 /data/dfs/ ext4 defaults,noatime 0 1  UUID=1f59e7a0-5ca8-4365-8e26-6d8b097edaa0 /var/lib/pgsql ext4 defaults,noatime,discard,barrier=0 0 1 |

Check disk performance

|  |
| --- |
| sudo hdparm -t /dev/sda |

## DataNode

The following tasks are executed to setup a DataNode

* Setup disks where HDFS will be configure. We follow the same procedure as described on the previous section

|  |
| --- |
| [root@interncloudera-worker02 clouderaadmin]# lsblk  NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT  sdc 8:32 0 512G 0 disk  sdd 8:48 0 512G 0 disk |

Disk format:

|  |
| --- |
| sudo mke2fs -F -t ext4 -b 4096 -E lazy\_itable\_init=1 -O sparse\_super,dir\_index,extent,has\_journal,uninit\_bg -m1 /dev/sdd |

Mount the disks on the following paths:

* /data/1/dfs/dn
* /data/2/dfs/dn
* /data/N/dfs/dn

## Tune the OS

Tune OS as Cloudera recomendation:

<https://github.com/Azure/azure-quickstart-templates/blob/master/cloudera-on-centos/scripts/initialize-node.sh>

# Disable the need for a tty when running sudo and allow passwordless sudo for the admin user

# Create Impala scratch directory (No s’ha configurat res)

# Disable SELinux

# Disable iptables

* Look as they are not installed
* On CentOS 7 seems they have been replaced by systemctl status firewalld

# Install and start NTP

|  |
| --- |
| sudo yum install ntp  timedatectl  sudo service ntpd start  sudo service ntpd status  sudo chkconfig ntpd on  sudo systemctl enable ntpd  timedatectl set-timezone Europe/Madrid  chkconfig ntpd  timedatectl |

# Disable THP (HugePages)

# Set swappiness to 1

# Set system tuning params

# Set host FQDN (no he configurat res)

|  |
| --- |
| **//Command per a veure la configuració del hostname:**  hostnamectl status  hostname -f  hostname -i |

## Afegir el Repositori del Cloudera

Setup Cloudera repository on the node where Cloudera Manager will be installed:

|  |
| --- |
| **// Descarreguem el repositori de Cloudera**  wget https://archive.cloudera.com/cm5/redhat/7/x86\_64/cm/cloudera-manager.repo  mv cloudera-manager.repo /etc/yum.repos.d/ |

# MySQL Database installation

The following link shows how to setup the DB

https://www.cloudera.com/documentation/enterprise/latest/topics/cm\_ig\_mysql.html

|  |
| --- |
| **//Repo**  wget http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm  sudo rpm -ivh mysql-community-release-el7-5.noarch.rpm  sudo yum update  sudo yum install mysql-community-server //instalem la versió community del mysql  **//DB startup**  [clouderaadmin@interncloudera-master-01 ~]$ **sudo systemctl start mysqld.service**  [clouderaadmin@interncloudera-master-01 ~]$  [clouderaadmin@interncloudera-master-01 ~]$ **sudo systemctl status mysqld.service**  ● mysqld.service - MySQL Community Server  Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)  Active: active (running) since Fri 2018-06-15 09:43:00 CEST; 6s ago  Process: 5856 ExecStartPost=/usr/bin/mysql-systemd-start post (code=exited, status=0/SUCCESS)  Process: 5842 ExecStartPre=/usr/bin/mysql-systemd-start pre (code=exited, status=0/SUCCESS)  Main PID: 5855 (mysqld\_safe)  CGroup: /system.slice/mysqld.service  ├─5855 /bin/sh /usr/bin/mysqld\_safe --basedir=/usr  └─6022 /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/usr/lib64/mysql/plugin --log-error=/var/log/mysqld.log --p...  Jun 15 09:42:59 interncloudera-master-01 systemd[1]: Starting MySQL Community Server...  Jun 15 09:42:59 interncloudera-master-01 mysqld\_safe[5855]: 180615 09:42:59 mysqld\_safe Logging to '/var/log/mysqld.log'.  Jun 15 09:42:59 interncloudera-master-01 mysqld\_safe[5855]: 180615 09:42:59 mysqld\_safe Starting mysqld daemon with databases from /var/lib/mysql  Jun 15 09:43:00 interncloudera-master-01 systemd[1]: Started MySQL Community Server.  [clouderaadmin@interncloudera-master-01 ~]$  **//Check version**  [clouderaadmin@interncloudera-master-01 ~]$ mysql -V  mysql Ver 14.14 Distrib 5.6.40, for Linux (x86\_64) using EditLine wrapper  **//Secure DB**  [clouderaadmin@interncloudera-master-01 ~]$ mysql\_secure\_installation |

MySQL password: 8S5F9dXDym9PUyss

To setup the replication, we have to activate the binary logging

Tune DB following Cloudera recomendations:

**my.cnf** File:

|  |
| --- |
| [mysqld]  transaction-isolation = READ-COMMITTED  # Disabling symbolic-links is recommended to prevent assorted security risks;  # to do so, uncomment this line:  # symbolic-links = 0  key\_buffer\_size = 32M  max\_allowed\_packet = 32M  thread\_stack = 256K  thread\_cache\_size = 64  query\_cache\_limit = 8M  query\_cache\_size = 64M  query\_cache\_type = 1  max\_connections = 550  #expire\_logs\_days = 10  #max\_binlog\_size = 100M  #log\_bin should be on a disk with enough free space. Replace '/var/lib/mysql/mysql\_binary\_log' with an appropriate path for your system  #and chown the specified folder to the mysql user.  log\_bin=/var/lib/mysql/mysql\_binary\_log  # For MySQL version 5.1.8 or later. For older versions, reference MySQL documentation for configuration help.  binlog\_format = mixed  read\_buffer\_size = 2M  read\_rnd\_buffer\_size = 16M  sort\_buffer\_size = 8M  join\_buffer\_size = 8M  # InnoDB settings  innodb\_file\_per\_table = 1  innodb\_flush\_log\_at\_trx\_commit = 2  innodb\_log\_buffer\_size = 64M  innodb\_buffer\_pool\_size = 4G  innodb\_thread\_concurrency = 8  innodb\_flush\_method = O\_DIRECT  innodb\_log\_file\_size = 512M  [mysqld\_safe]  log-error=/var/log/mysqld.log  pid-file=/var/run/mysqld/mysqld.pid  sql\_mode=STRICT\_ALL\_TABLES |

Check mysql will start on boot:

|  |
| --- |
| sudo /sbin/chkconfig mysqld on  Note: Forwarding request to 'systemctl enable mysqld.service'.  systemctl status mysqld.service  ● mysqld.service - MySQL Community Server  Loaded: loaded (/usr/lib/systemd/system/mysqld.service; **enabled**; vendor preset: disabled)  Active: inactive (dead) since Fri 2018-06-15 10:15:36 CEST; 8min ago |

Start MySQL service:

|  |
| --- |
| [clouderaadmin@interncloudera-master-01 ~]$ sudo service mysqld status  Redirecting to /bin/systemctl status mysqld.service  ● mysqld.service - MySQL Community Server  Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)  Active: active (running) since Fri 2018-06-15 10:26:05 CEST; 5s ago  Process: 8229 ExecStartPost=/usr/bin/mysql-systemd-start post (code=exited, status=0/SUCCESS)  Process: 8214 ExecStartPre=/usr/bin/mysql-systemd-start pre (code=exited, status=0/SUCCESS)  Main PID: 8228 (mysqld\_safe)  CGroup: /system.slice/mysqld.service  ├─8228 /bin/sh /usr/bin/mysqld\_safe --basedir=/usr  └─8615 /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql --plugin-dir=/usr/lib64/mysql/plugin --log-error=/var/log/mysqld.log --p...  Jun 15 10:25:52 interncloudera-master-01 systemd[1]: Starting MySQL Community Server...  Jun 15 10:25:52 interncloudera-master-01 mysqld\_safe[8228]: 180615 10:25:52 mysqld\_safe Logging to '/var/log/mysqld.log'.  Jun 15 10:25:52 interncloudera-master-01 mysqld\_safe[8228]: 180615 10:25:52 mysqld\_safe Starting mysqld daemon with databases from /var/lib/mysql  Jun 15 10:26:05 interncloudera-master-01 systemd[1]: Started MySQL Community Server. |

Restart the VM to ensure everything works as expected

# HA setup on the Database

https://www.cloudera.com/documentation/enterprise/5-6-x/topics/admin\_cm\_ha\_dbms.html

Check setup on Node#1

|  |
| --- |
| **// Bin set a 1**  mysql> select variable\_value as "BINARY LOGGING STATUS (log-bin) :: "  -> from information\_schema.global\_variables where variable\_name='log\_bin';  +-------------------------------------+  | BINARY LOGGING STATUS (log-bin) :: |  +-------------------------------------+  | ON |  +-------------------------------------+  1 row in set (0.00 sec)  **//Check Server ID value that has to be unique across all nodes. If not set it up to 1**  mysql> select variable\_value as "SERVER ID " from information\_schema.global\_variables where variable\_name='server\_id';  +------------+  | SERVER ID |  +------------+  | 1 |  +------------+  1 row in set (0.00 sec) |

On the Slave Node:

https://dev.mysql.com/doc/refman/8.0/en/replication-howto-slavebaseconfig.html

|  |
| --- |
| wget http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm  sudo rpm -ivh mysql-community-release-el7-5.noarch.rpm  sudo yum update  sudo yum install mysql-community-server //instalem la versió community del mysql  mysql -V  mysql\_secure\_installation |

Modify the my.cnf file with the same configuration as the Master node.

Setup the “server-id” 2 on the Slave node, and we run a restart. It has to be a different value than Master node:

|  |
| --- |
| [mysqld]  server-id=2 |

Check server-ID value:

mysql> select variable\_value as "SERVER ID " from information\_schema.global\_variables where variable\_name='log\_bin';

Node MASTER:

Setup Master node for Replication user:

|  |
| --- |
| mysql> CREATE USER 'repl'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss';  mysql> GRANT REPLICATION SLAVE ON \*.\* TO 'repl'@'%';  **//Comprovem l’usuari creat:**  SELECT User, Host, Password FROM mysql.user; |

Snapshot

|  |
| --- |
| mysql> SHOW MASTER STATUS;  +-------------------------+----------+--------------+------------------+-------------------+  | File | Position | Binlog\_Do\_DB | Binlog\_Ignore\_DB | Executed\_Gtid\_Set |  +-------------------------+----------+--------------+------------------+-------------------+  | mysql\_binary\_log.000006 | 252373 | | | |  +-------------------------+----------+--------------+------------------+-------------------+  1 row in set (0.00 sec)  mysql> FLUSH TABLES WITH READ LOCK;  **//Prenc el DUMP**  mysqldump --all-databases --master-data > dbdump.db  **//Deixo la sessio corrent per assegurar que no es fan canvis a la DDBB mentre prenc el snapshot** |

On a new session run:

|  |
| --- |
| mysqldump -u root -p --all-databases --master-data > dbdump.db |

A dbdump.db file is generated

Under the slave node, import the DB

|  |
| --- |
| mysql -u root -p < dbdump.db |

On the master node:

|  |
| --- |
| mysql> UNLOCK TABLES; |

On the Slave node modify the file /etc/my.cnf to ensure it starts:

|  |
| --- |
| [mysqld]  skip-slave-start |

Restart slave node to ensure it takes the slave role  
Setup Slave node to connect to Master node with the following information:

https://dev.mysql.com/doc/refman/8.0/en/replication-howto-slaveinit.html

|  |
| --- |
| CHANGE MASTER TO  MASTER\_HOST='interncloudera-master-01.local',  MASTER\_USER='repl',  MASTER\_PASSWORD='8S5F9dXDym9PUyss',  MASTER\_LOG\_FILE='mysql\_binary\_log.000006',  MASTER\_LOG\_POS=252373; //Aquí hi posem el valor que tenia quan hem fet el LOCK |

|  |
| --- |
| **//Start slave Node**  mysql> START SLAVE;  **//If everything is running as expected, Slave should be connected to the Master**  mysql> SHOW SLAVE STATUS\G;  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Slave\_IO\_State: Waiting for master to send event  Master\_Host: interncloudera-master-01.local  Master\_User: repl  Master\_Port: 3306  Connect\_Retry: 60  Master\_Log\_File: mysql\_binary\_log.000007  Read\_Master\_Log\_Pos: 75308  Relay\_Log\_File: mysqld-relay-bin.000002  Relay\_Log\_Pos: 1403  Relay\_Master\_Log\_File: mysql\_binary\_log.000007  Slave\_IO\_Running: **Yes**  Slave\_SQL\_Running: **Yes**  Replicate\_Do\_DB:  Replicate\_Ignore\_DB:  Replicate\_Do\_Table:  Replicate\_Ignore\_Table:  Replicate\_Wild\_Do\_Table:  Replicate\_Wild\_Ignore\_Table:  Last\_Errno: 0  Last\_Error:  Skip\_Counter: 0  Exec\_Master\_Log\_Pos: 75308  Relay\_Log\_Space: 1577  Until\_Condition: None  Until\_Log\_File:  Until\_Log\_Pos: 0  Master\_SSL\_Allowed: No  Master\_SSL\_CA\_File:  Master\_SSL\_CA\_Path:  Master\_SSL\_Cert:  Master\_SSL\_Cipher:  Master\_SSL\_Key:  Seconds\_Behind\_Master: 0  Master\_SSL\_Verify\_Server\_Cert: No  Last\_IO\_Errno: 0  Last\_IO\_Error:  Last\_SQL\_Errno: 0  Last\_SQL\_Error:  Replicate\_Ignore\_Server\_Ids:  Master\_Server\_Id: 1  Master\_UUID: 338131de-706a-11e8-ba8f-000d3a2c953f  Master\_Info\_File: /var/lib/mysql/master.info  SQL\_Delay: 0  SQL\_Remaining\_Delay: NULL  Slave\_SQL\_Running\_State: Slave has read all relay log; waiting for the slave I/O thread to update it  Master\_Retry\_Count: 86400  Master\_Bind:  Last\_IO\_Error\_Timestamp:  Last\_SQL\_Error\_Timestamp:  Master\_SSL\_Crl:  Master\_SSL\_Crlpath:  Retrieved\_Gtid\_Set:  Executed\_Gtid\_Set:  Auto\_Position: 0 |

Under the master we check the configuration:

|  |
| --- |
| mysql> SHOW SLAVE HOSTS;  +-----------+------+------+-----------+--------------------------------------+  | Server\_id | Host | Port | Master\_id | Slave\_UUID |  +-----------+------+------+-----------+--------------------------------------+  | 2 | | 3306 | 1 | 206ff4b8-72f6-11e8-8b2b-000d3a238034 |  +-----------+------+------+-----------+--------------------------------------+  1 row in set (0.00 sec) |

Modifty the file /etc/my.cnf on the Slave node:

|  |
| --- |
| [mysqld]  skip-slave-start |

Restart Slave

|  |
| --- |
| mysql> SHOW SLAVE STATUS\G;  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Slave\_IO\_State: Waiting for master to send event  Master\_Host: interncloudera-master-01.local  Master\_User: repl  Master\_Port: 3306  Connect\_Retry: 60  Master\_Log\_File: mysql\_binary\_log.000033  Read\_Master\_Log\_Pos: 4274993  Relay\_Log\_File: mysqld-relay-bin.000071  Relay\_Log\_Pos: 4275163  Relay\_Master\_Log\_File: mysql\_binary\_log.000033  **Slave\_IO\_Running: Yes**  **Slave\_SQL\_Running: Yes** |

# Install MySQL JDBC Driver

<https://www.cloudera.com/documentation/enterprise/latest/topics/cm_ig_mysql.html#cmig_topic_5_5_3>

It is recommended to not install it via yum

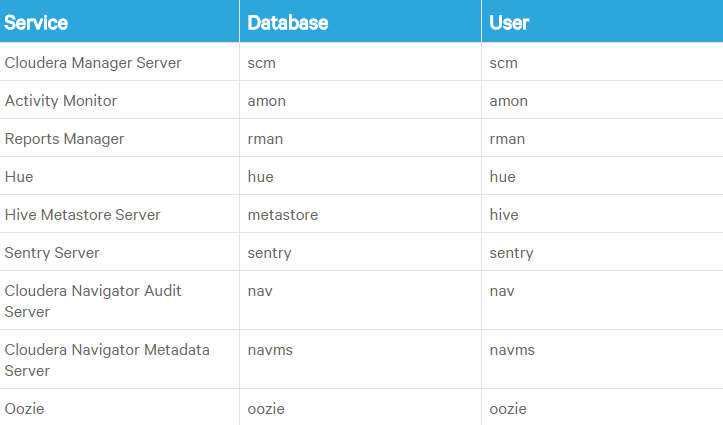
We create the directory /usr/share/java, where the jar file is copied

|  |
| --- |
| sudo cp /home/clouderaadmin/mysql-connector-java-5.1.46-bin.jar /usr/share/java/mysql-connector-java.jar |

# Create DB Tables

Setup initial tables on the DB

<https://www.cloudera.com/documentation/enterprise/latest/topics/cm_ig_mysql.html#concept_dsg_3mq_bl>



|  |
| --- |
| $> mysql -u root -p  $mysql>  CREATE DATABASE cmserver DEFAULT CHARACTER SET utf8;  GRANT ALL on cmserver.\* TO 'cmserveruser'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss';  CREATE DATABASE amon DEFAULT CHARACTER SET utf8;  GRANT ALL on amon.\* TO 'amon'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss';  CREATE DATABASE rman DEFAULT CHARACTER SET utf8;  GRANT ALL on rman.\* TO 'rman'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss';  CREATE DATABASE metastore DEFAULT CHARACTER SET utf8;  GRANT ALL on metastore.\* TO 'hive'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss';  CREATE DATABASE sentry DEFAULT CHARACTER SET utf8;  GRANT ALL on sentry.\* TO 'sentry'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss';  CREATE DATABASE nav DEFAULT CHARACTER SET utf8;  GRANT ALL on nav.\* TO 'nav'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss';  CREATE DATABASE navms DEFAULT CHARACTER SET utf8;  GRANT ALL on navms.\* TO 'navms'@'%' IDENTIFIED BY '8S5F9dXDym9PUyss'; |

We reuse the same root password

# DataBase HUE i OOZIE

Hue table:

https://www.cloudera.com/documentation/enterprise/latest/topics/hue\_dbs\_mysql.html#concept\_tq4\_tbt\_zw

|  |
| --- |
| create database hue default character set utf8 default collate utf8\_general\_ci;  grant all on hue.\* to 'hue'@'%' identified by '8S5F9dXDym9PUyss';  select \* from information\_schema.schemata;  quit |

Hi ha uns comandos per a assegurar que Hue està correctament configurada, però no els executo. Son els que apareixen a la Note:

|  |
| --- |
| create database oozie default character set utf8;  grant all privileges on oozie.\* to 'oozie'@'localhost' identified by '8S5F9dXDym9PUyss';  grant all privileges on oozie.\* to 'oozie'@'%' identified by '8S5F9dXDym9PUyss'; |

Pending to apply the following configuration:

https://www.cloudera.com/documentation/enterprise/latest/topics/cm\_ig\_mysql.html#id\_ijy\_cwt\_g5\_\_section\_fvr\_bxt\_jp

# Install Oracle JDK

We use JDK 1.8u144 version as it is the latest supported by Cloudera

We download it from Jave webiste.

Oracle JDK has to be installed on all the VMs

<http://www.oracle.com/technetwork/java/javase/downloads/java-archive-javase8-2177648.html>

*“ Java Platform: Standard Edition (SE), Enterprise Edition (EE), and Micro Edition (ME). This tutorial is focused on Java SE (Java Platform, Standard Edition).There are two different Java SE packages that can be installed: the Java Runtime Environment (JRE) and the Java Development Kit (JDK). JRE is an implementation of the Java Virtual Machine (JVM), which allows you to run compiled Java applications and applets. JDK includes JRE and other software that is required for writing, developing, and compiling Java applications and applets. There are also two different implementations of Java: OpenJDK and Oracle Java. Both implementations are based largely on the same code but OpenJDK, the reference implementation of Java, is fully open source while Oracle Java contains some proprietary code”*

|  |
| --- |
| sudo yum localinstall jdk-8u144-linux-x64.rpm  [clouderaadmin@interncloudera-master-01 ~]$ java -version  java version "1.8.0\_144"  Java(TM) SE Runtime Environment (build 1.8.0\_144-b01)  Java HotSpot(TM) 64-Bit Server VM (build 25.144-b01, mixed mode)  sudo vi /etc/profile.d/set-java.sh  export JAVA\_HOME=/usr/java/latest  export PATH=$JAVA\_HOME/bin:$PATH  export CLASSPATH=/usr/share/java:/usr/java/latest/lib:$CLASSPATH  sudo vi /etc/profile.d/set-java.csh  setenv JAVA\_HOME /usr/java/latest  setenv PATH $JAVA\_HOME/bin:$PATH  setenv CLASSPATH /usr/share/java:/usr/java/latest/lib:$CLASSPATH |

Check environment variables are setup as expected:

|  |
| --- |
| JAVA\_HOME=/usr/java/latest  CLASSPATH=/usr/share/java:/usr/java/latest/lib:  PATH=/usr/java/latest/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/home/clouderaadmin/.local/bin:/home/clouderaadmin/bin |

# Instal.lar els paquets del Cloudera Manager

Install the following packages:

|  |
| --- |
| sudo yum install cloudera-manager-daemons cloudera-manager-server |

Access the Portal

<http://40.113.154.185:7180>

# Configurar el Cloudera Manager Server Database

Guide to setup the Database

<https://www.cloudera.com/documentation/enterprise/latest/topics/cm_ig_installing_configuring_dbs.html#cmig_topic_5_2>

Start Cloudera Manager

|  |
| --- |
| sudo service cloudera-scm-server start |

Following Oscar recomendation, we tune the file:

|  |
| --- |
| # Specify any command line arguments for the Cloudera SCM Server here.  #  CMF\_SERVER\_ARGS=""  #  # Locate the JDBC driver jar file.  #  # The default value is the default system mysql driver on RHEL/CentOS/Ubuntu  # and the standard, documented location for where to put the oracle jar in CM  # deployments.  #  export CMF\_JDBC\_DRIVER\_JAR="/usr/share/java/mysql-connector-java.jar:/usr/share/java/oracle-connector-java.jar"  #  # Java Options.  #  # Default value sets Java maximum heap size to 2GB, and Java maximum permanent  # generation size to 256MB.  #  **export CMF\_JAVA\_OPTS="-Xmx4G -XX:MaxPermSize=512m -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/tmp"** |

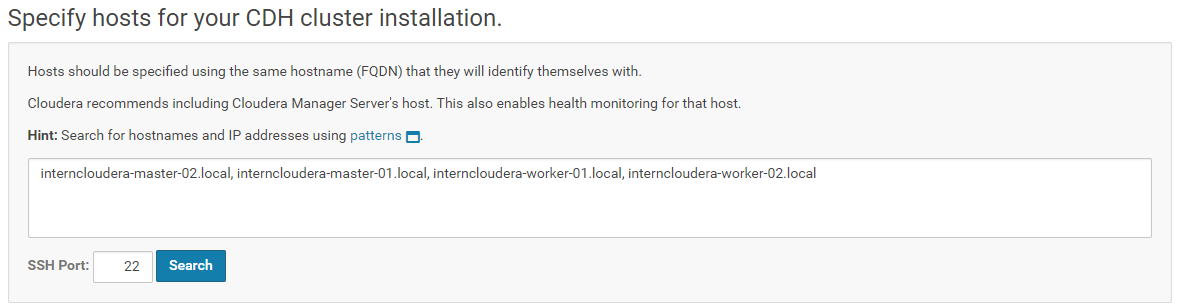
Restart Cloudera Manager

Cloudera Manager should be available on:

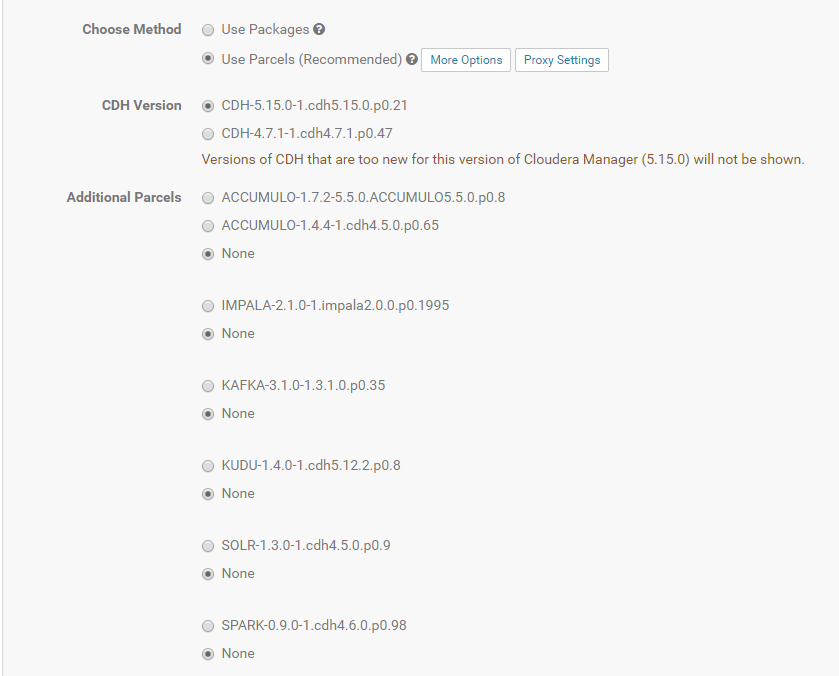
<http://51.144.42.202:7180/>

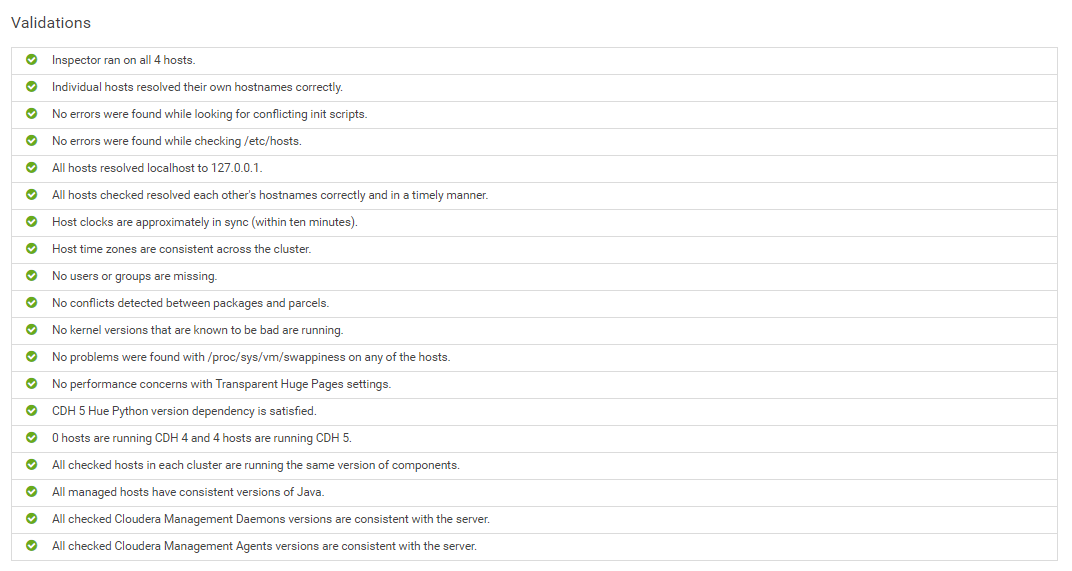
# Install CDH on Cloudera Manager Server Database

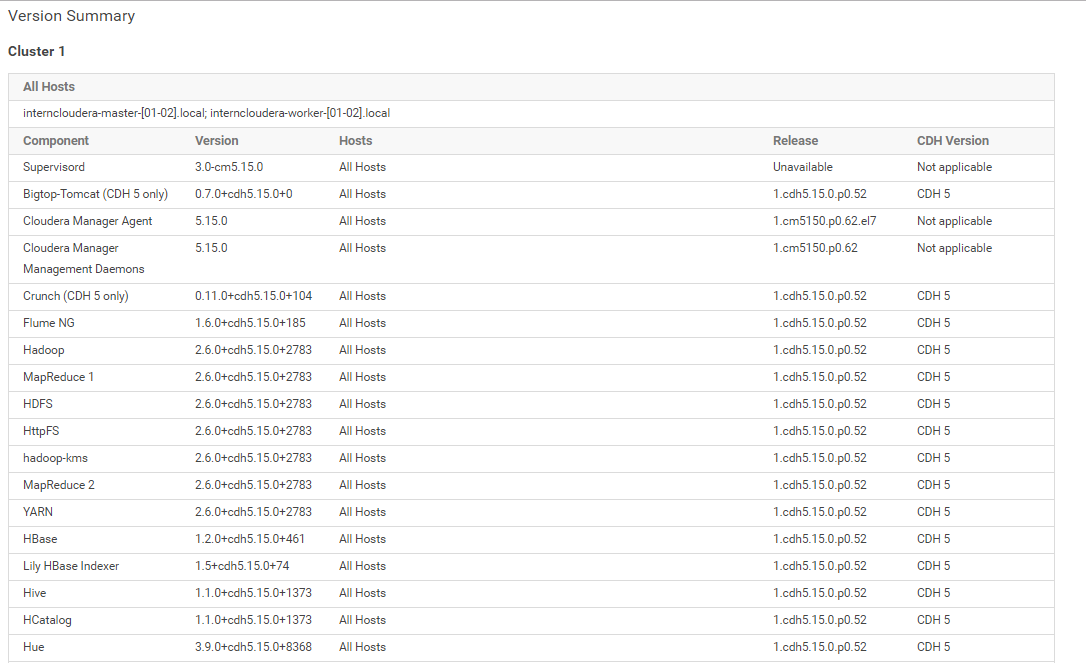
Identify the rest of hosts where CDH will be install:

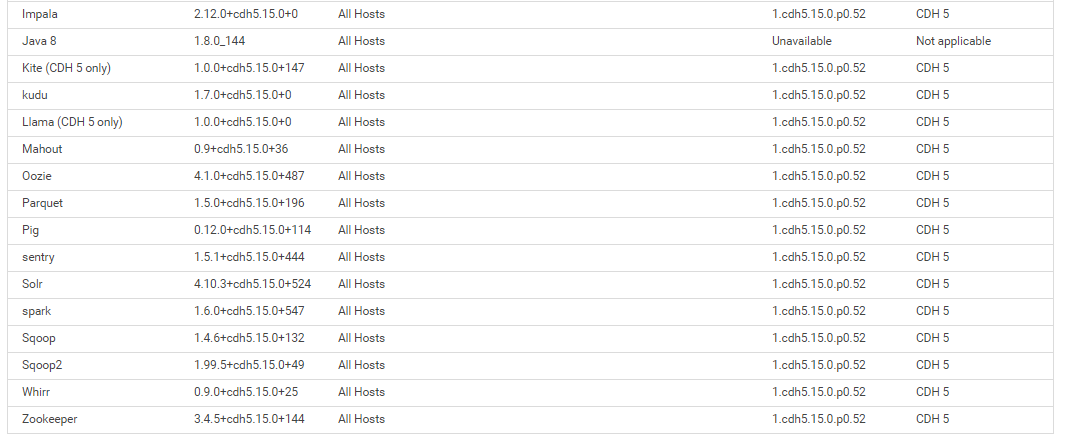


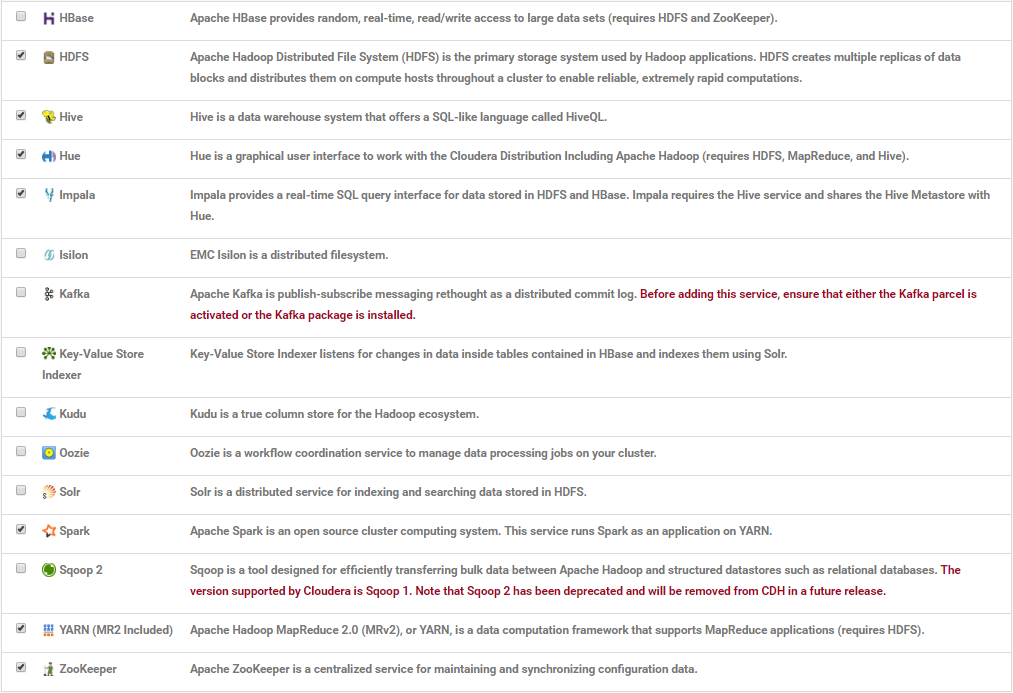
Escollim les opcions:

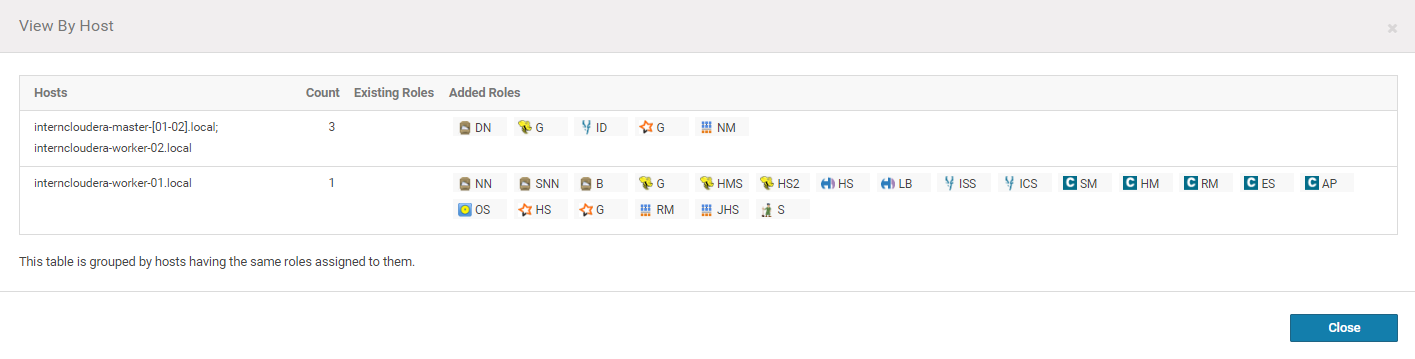


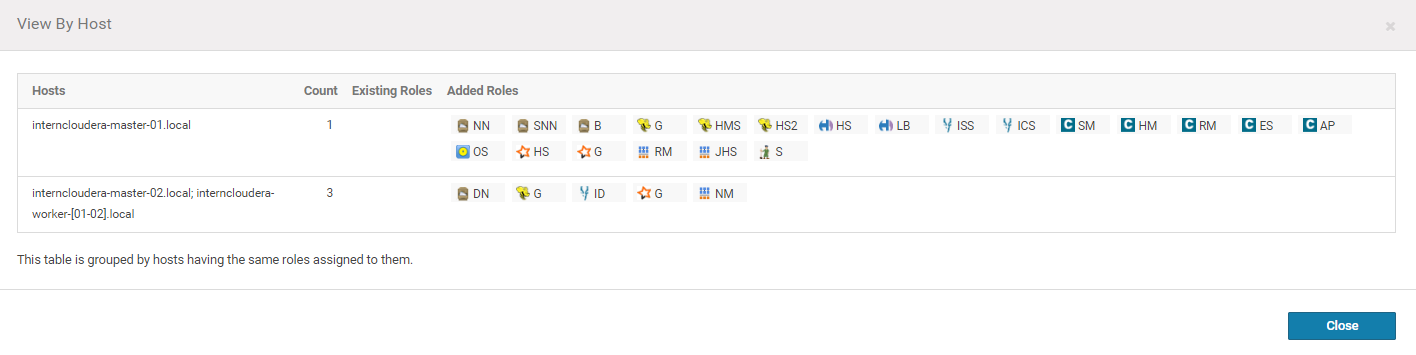




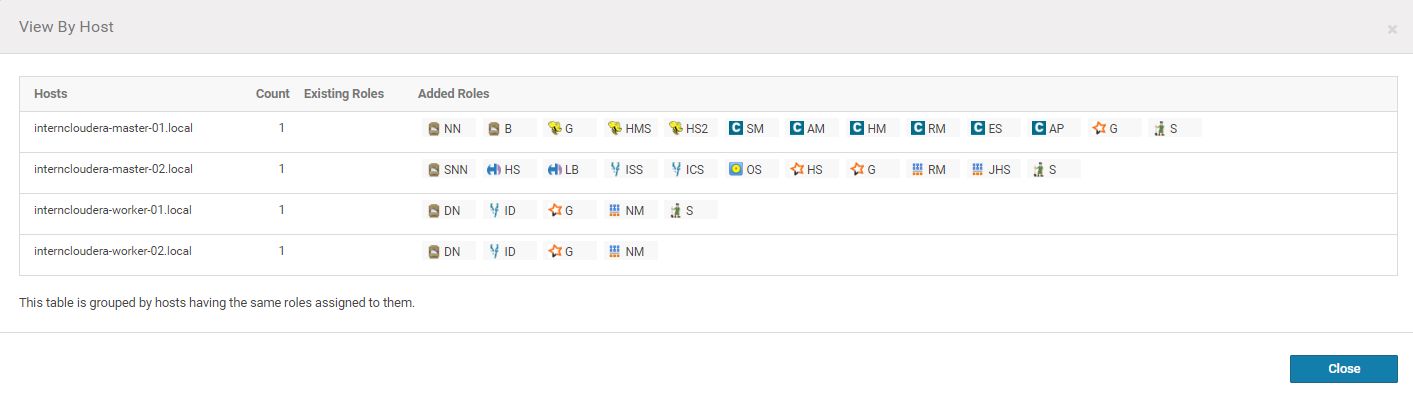


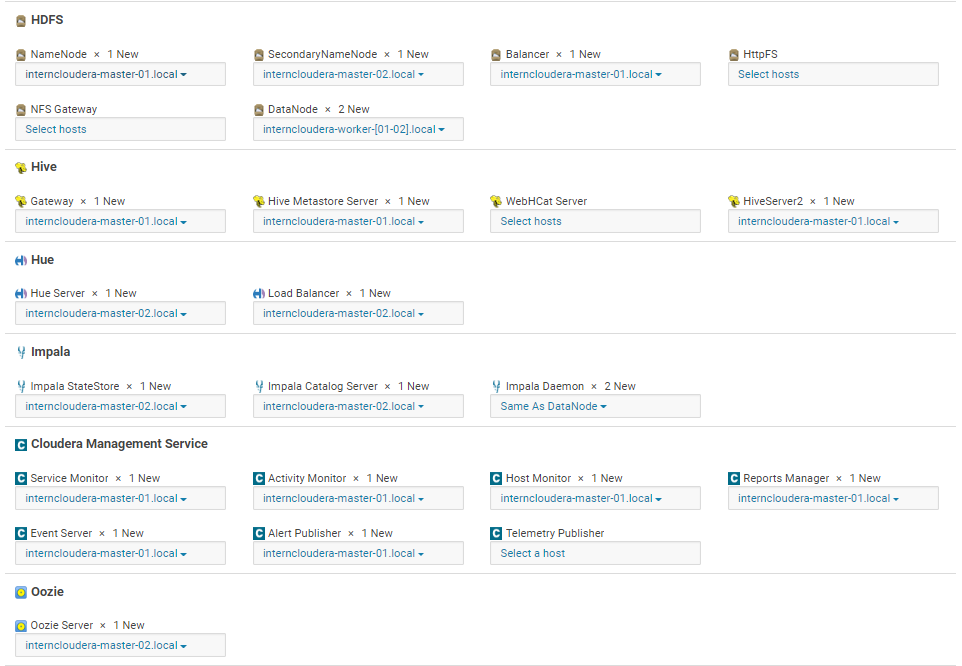


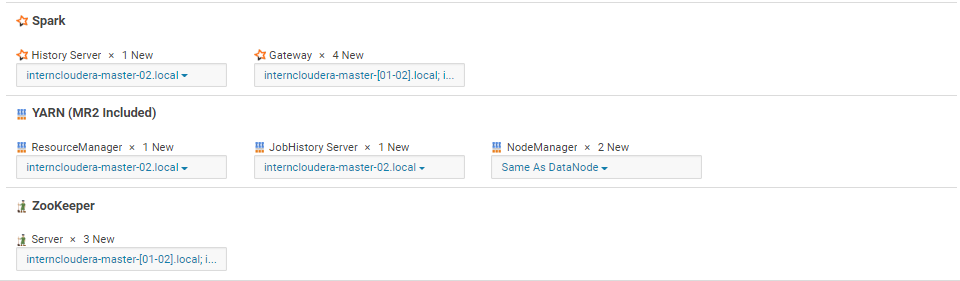


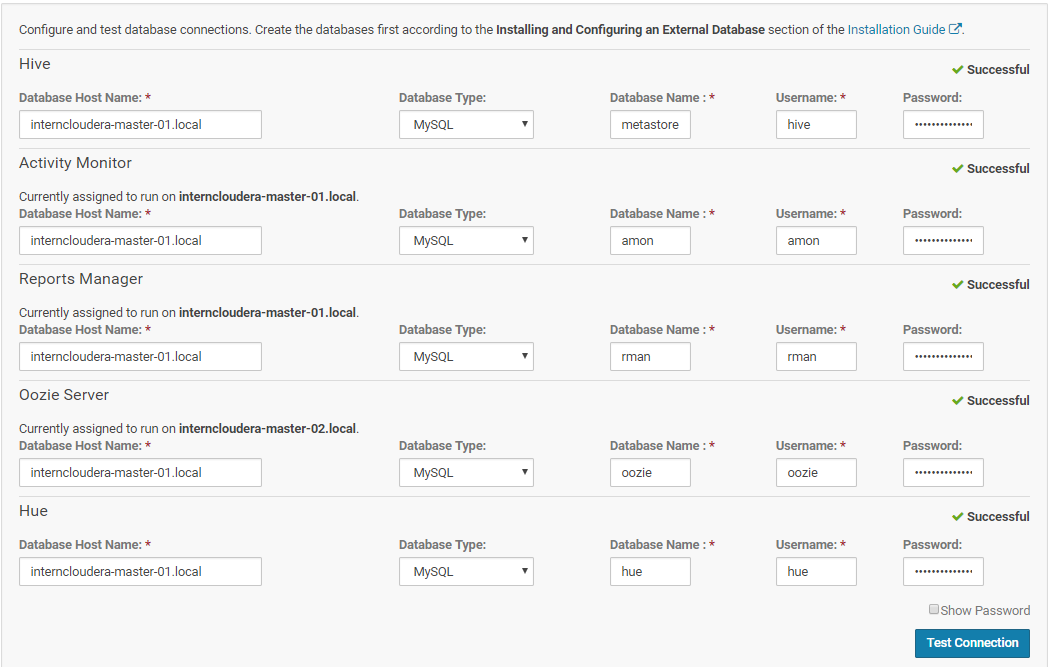


Configuració Final:









The following file gathers the information from Cloudera Manager:

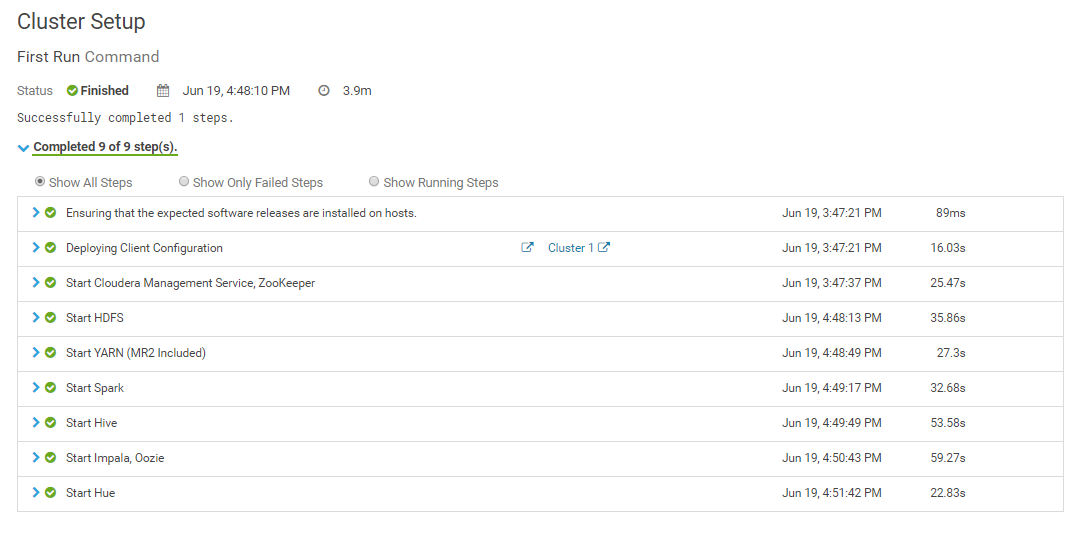
****

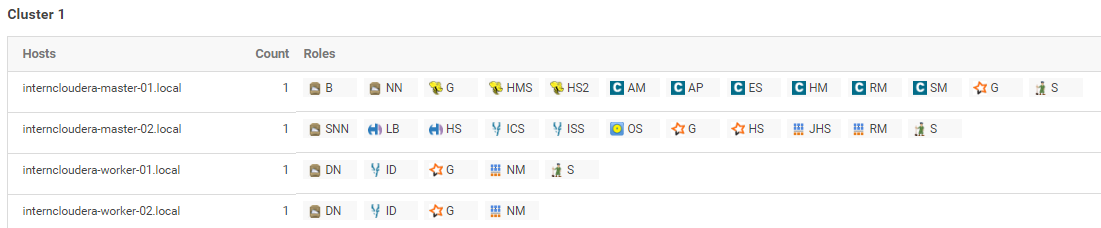
The following setup has been modified:

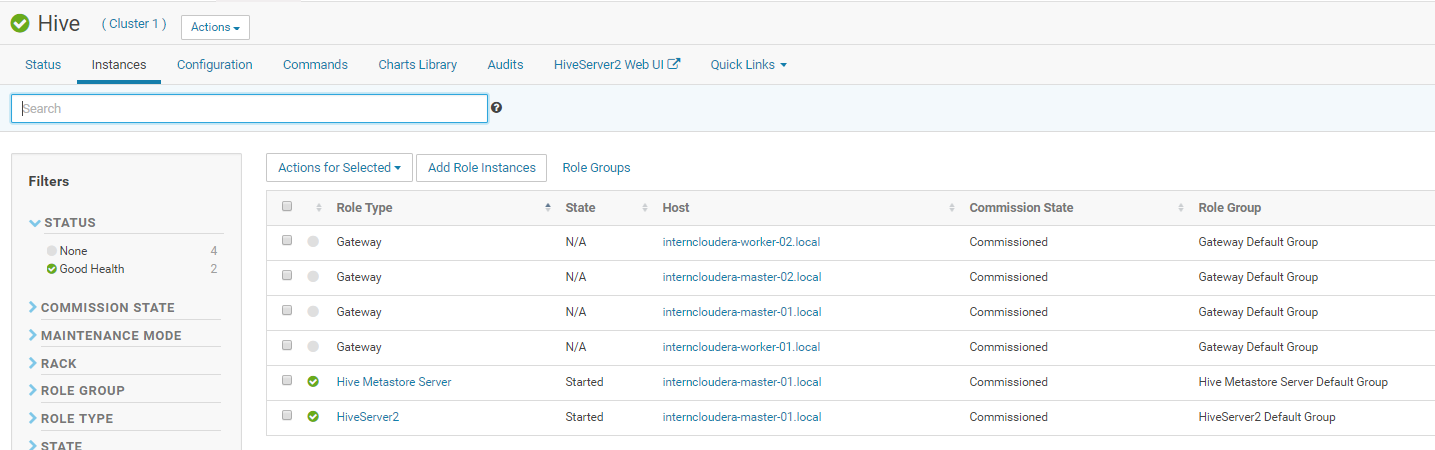
|  |
| --- |
| chmod 0755 /data/1/dfs/dn  chmod 0755 /data/2/dfs/dn  chown hdfs:hadoop /data/1/dfs/dn  chown hdfs:hadoop /data/2/dfs/dn |

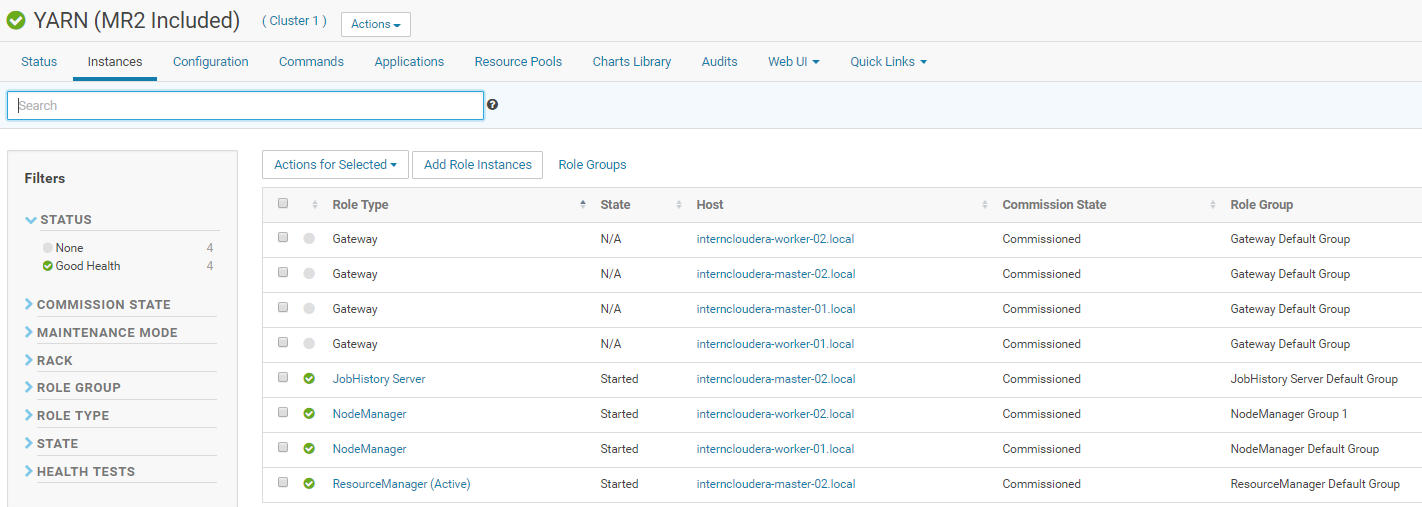
|  |
| --- |
| sudo chown -R clouderaadmin:clouderaadmin /data  chmod 777 -R /data |

Cluster has been successfully deployed:









# Configurar HDFS HA i Automatic Failover

Following Cloudera recomendations, we setup HDFS HA and Automatic Failover

*Note: In an HA cluster, the standby NameNode also performs checkpoints of the namespace state, and thus it is not necessary to run a Secondary NameNode, CheckpointNode, or BackupNode in an HA cluster. In fact, to do so would be an error. If you are reconfiguring a non-HA-enabled HDFS cluster to be HA-enabled, you can reuse the hardware which you had previously dedicated to the Secondary NameNode.*

We make use of the following services:

* Quorum Journals
* Zookeeper and ZKFailoverController

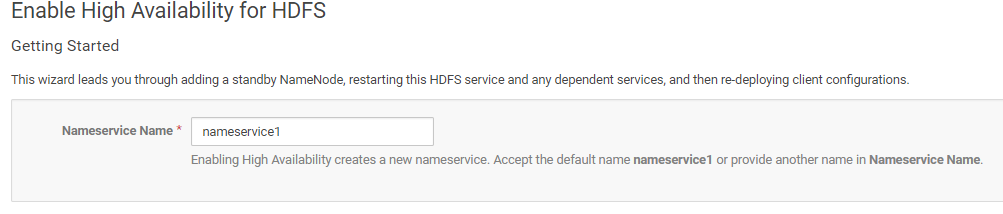
First Quorum Journals are setup.**JournalNode Edits**

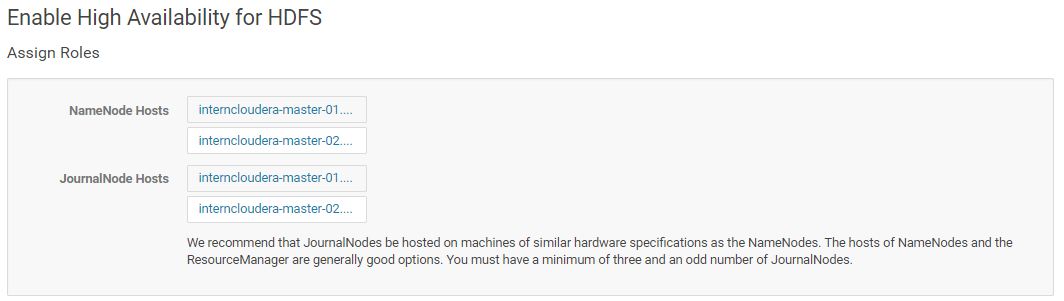
**Directory dfs.journalnode.edits.dir**

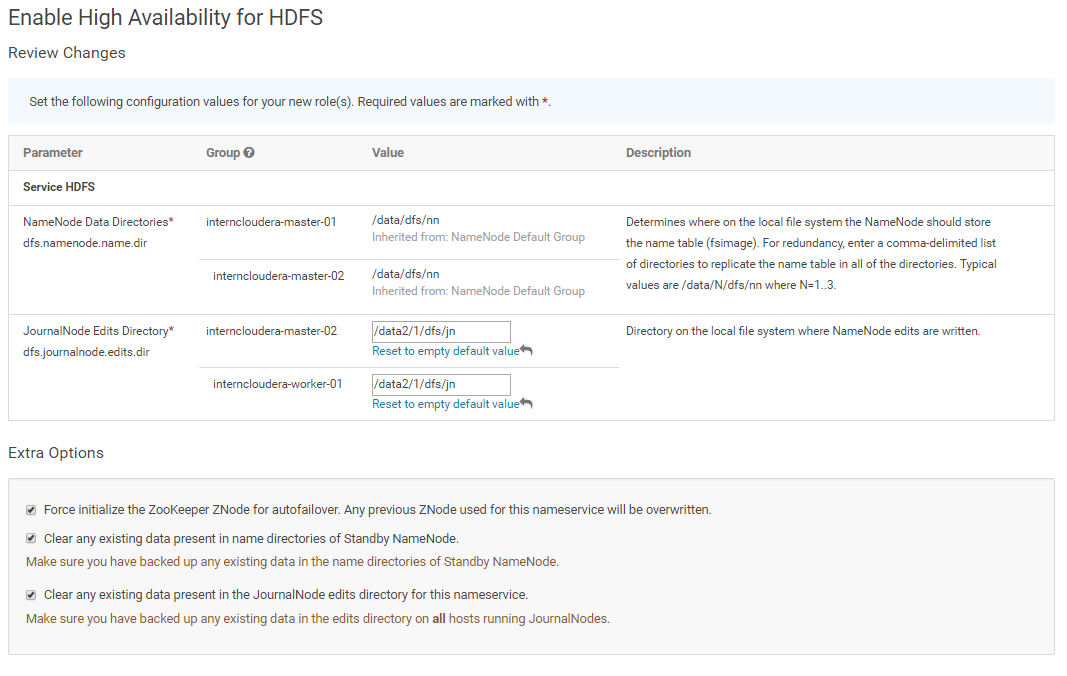
|  |
| --- |
| /data/dfs/nn |

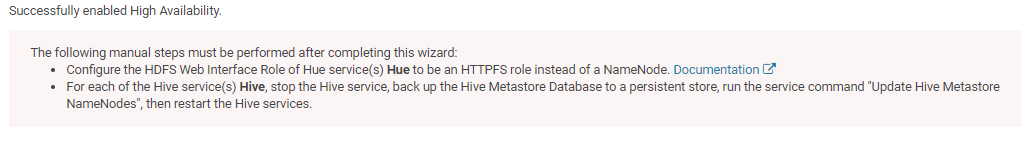
|  |
| --- |
| sudo mkdir -p /data2/1/dfs/jn  sudo chown -R hdfs:hdfs /data2/1/dfs/jn |

https://www.cloudera.com/documentation/cdh/5-0-x/CDH5-High-Availability-Guide/cdh5hag\_hdfs\_ha\_software\_config.html





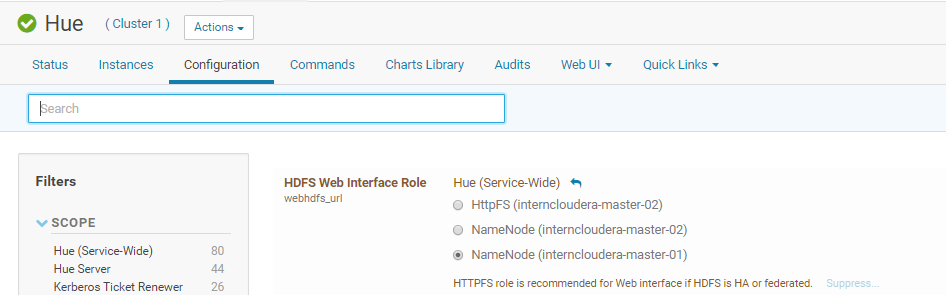




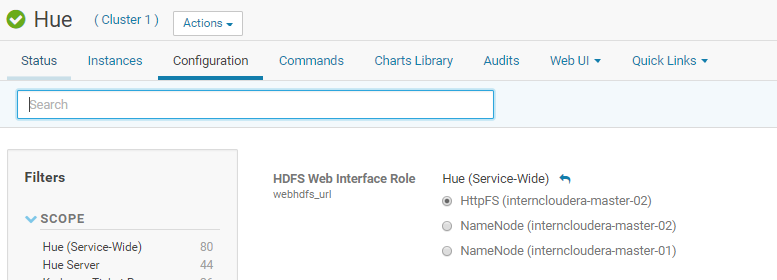
We have following this documentation:

<https://www.cloudera.com/documentation/enterprise/5-3-x/topics/cdh_hag_hdfs_ha_cdh_components_config.html#concept_rj1_hsq_bp>

We modifity Hue setup:

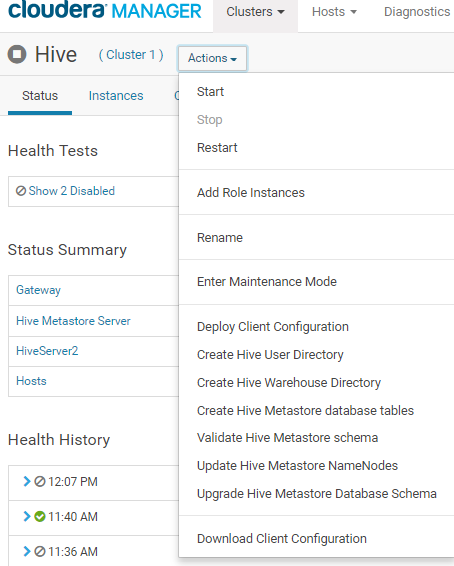


To:



To apply the second recomendation we execute the following steps:

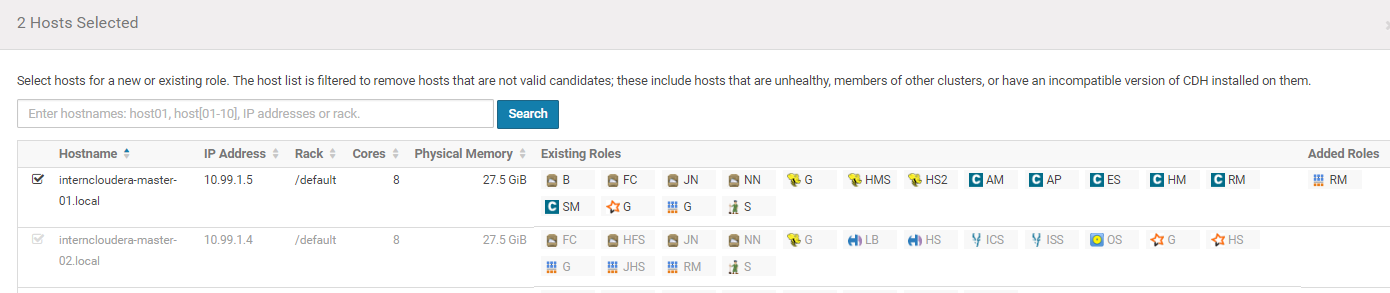
* Stop Hive service from Cloudera Manager
* Take a dump of BBDD: mysqldump -u root -p --databases metastore > /home/clouderaadmin/dump\_hive\_metastore.sql
* Run a “Update Hive Metastore”
* Service restart



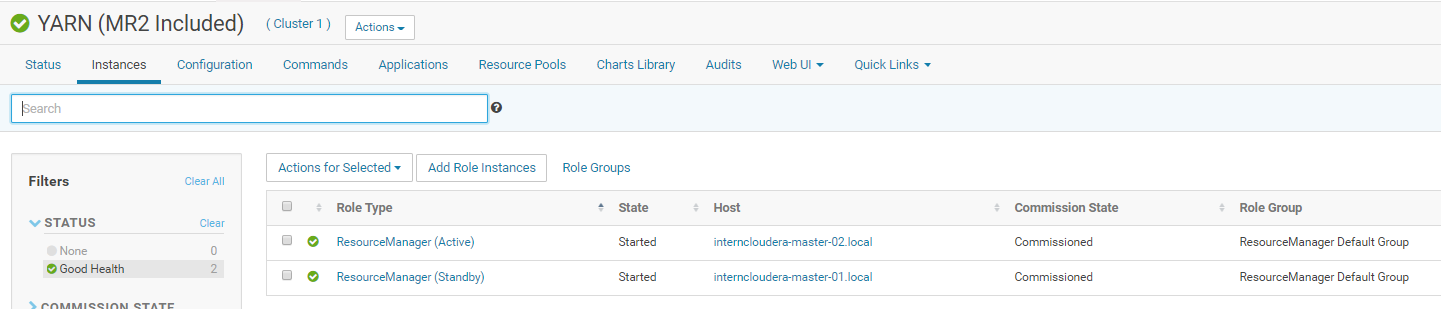
# Setup YARN High Availability

From Cloudera Manager portal, YARN we select Enable HA

Add role to Master 01



Check Active and Passive:

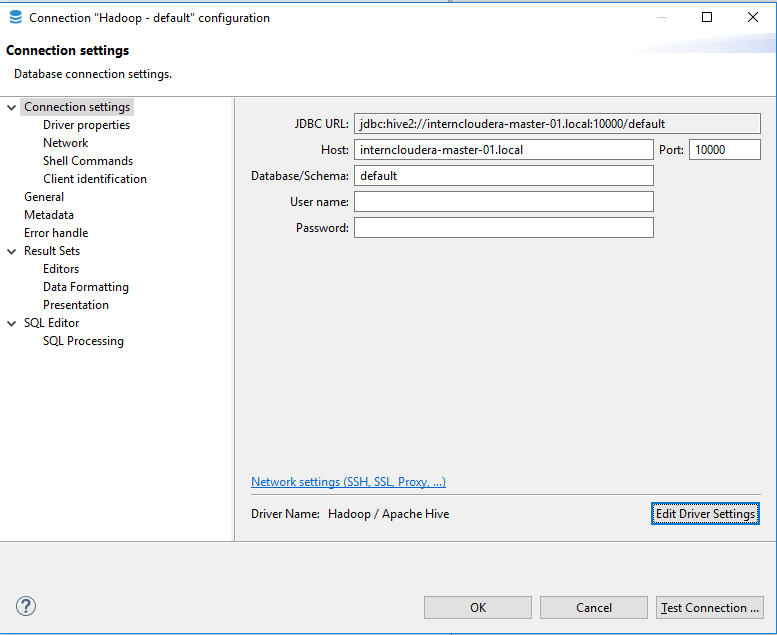


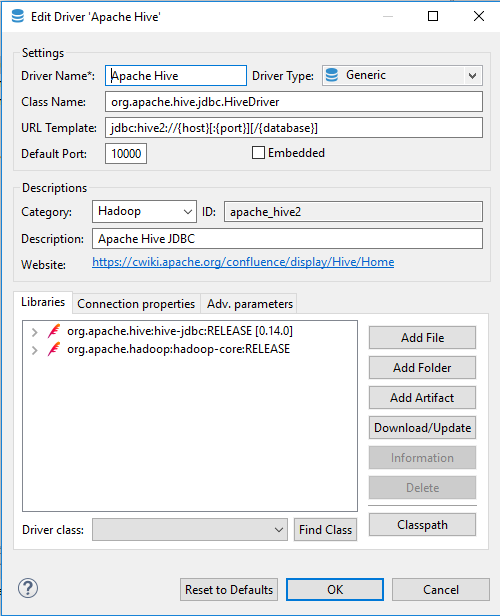
Links acces YARN:

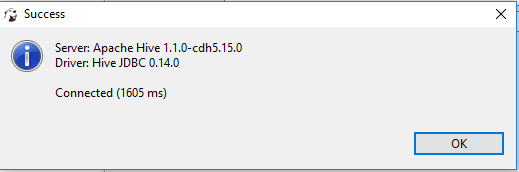
* Resource Manager Web UI: port 8088
* MapReduce JobHistory Server: port 19888
* Spark History: port 18088

# Connecting to Hive

Connect to Hive Metastore from Dbeaver:







# Private IP addressing and /etc/host file configuration

10.99.1.4 interncloudera-master-02.local

10.99.1.5 interncloudera-master-01.local

10.99.1.6 interncloudera-tools-01.local

10.99.1.7 interncloudera-worker-01.local

10.99.1.8 interncloudera-worker-02.local

10.99.1.9 interncloudera-tools-02.local



FOR MORE INFORMATION:

[info@clearpeaks.com](mailto:info@clearpeaks.com)  
[www.clearpeaks.com](http://www.clearpeaks.com)  
  
Spain: +34 93 272 154  
Abu Dhabi: +971 2 508 9130

**Proprietary and Confidential © 2017 ClearPeaks. All rights reserved.**

All information contained in this document is confidential and proprietary to ClearPeaks.

No part of this document may be photocopied, electronically transferred, modified, or reproduced in any manner without the prior written content of ClearPeaks.