**Cloudera Manager**

**CentOS7**

**離線安裝說明**

**章節**

1. **OverAll**
2. **Environment Validation**
3. **Server Configuration**
4. **Cloudera Manager Software download**
5. **Install Cloudera Manger**
6. **Install Hadoop CDH**
7. **Enable High Availability**
8. **Installed Validation**
9. **Enable Kerberos**
10. **Remark**
11. **OverAll**

此份文件主要介紹如何安裝Cloudera Manager，並透過Cloudera Manager Deploy Hadoop CDH 5.8，此次安裝前的先決條件

1. 離線安裝
2. 關閉防火牆
3. 5+1VM: Cluster共五台加一台ClouderaManager VM
4. DB使用mysql 5.7
5. OS: CentOS 7.2 x86\_64版本
6. 執行時不可使用root帳號

接下來會開始介紹安裝步驟，主要有下列步驟依序介紹：

1. Server 基本環境配置
2. Component下載並建立local repository
3. 安裝Cloudera Manager Server
4. 透過GUI安裝Agent及部署Hadoop Cluster
5. **Environment Validation**

若為全新環境安裝請跳過此步驟

1. check Linux version

|  |
| --- |
| #> pssh -h nodes.txt -i "uname -r" |

CDH所支援的Linux版本請參考<https://www.cloudera.com/documentation/enterprise/5-8-x/topics/cm_ig_cm_requirements.html#cmig_topic_4_1>

1. check FileSystem is ext3 or ext4

|  |
| --- |
| #> pssh -h nodes.txt -i "cat /etc/fstab" |

1. check java JDK version

|  |
| --- |
| #> pssh -h nodes.txt -i "java -version" |

CDH所支援的JDK版本請參考https://www.cloudera.com/documentation/enterprise/5-8-x/topics/cm\_ig\_cm\_requirements.html#concept\_tjs\_3rg\_dk

1. check ClouderaManager disk capacity

|  |
| --- |
| #> pssh -h nodes.txt -i "df -h" |

ClouderaManager所需使用的磁碟空間請參考:<http://www.cloudera.com/documentation/enterprise/release-notes/topics/rn_consolidated_pcm.html#concept_kr3_w13_hw>

1. check host name is FQDN

|  |
| --- |
| #> pssh -h nodes.txt -i "hostname"  #> pssh -h nodes.txt -i "cat /etc/hosts"  #> pssh -h nodes.txt -i "/etc/sysconfig/network" |

1. check network environment

|  |
| --- |
| #> pssh -h nodes.txt -i "/sbin/ifconfig"  #> pssh -h nodes.txt -i "ping cms.databox.com.tw"  #> pssh -h nodes.txt -i "ping cm1.databox.com.tw"  #> pssh -h nodes.txt -i "ping cm2.databox.com.tw"  …..  #> pssh -h nodes.txt -i "ping cm5.databox.com.tw" |

1. Permission Requirements with Cloudera Manager

sudo user command test

|  |
| --- |
| #> pssh -h nodes.txt -x “-tt” -i "sudo yum --version"  #> pssh -h nodes.txt -x “-tt” -i "sudo /sbin/chkconfig"  #> pssh -h nodes.txt -x “-tt” -i "sudo service network status"  ##待處理  mv rm chown id install sed |

1. Role Assignment reference:

<https://www.cloudera.com/documentation/enterprise/5-8-x/topics/cm_ig_host_allocations.html#concept_f43_j4y_dw>

1. **Server Configuration**

安裝CentOS 7.3 選擇最小安裝(minimal)來做為基本OS，六台VM基本的IP配置如下

|  |  |  |  |
| --- | --- | --- | --- |
| **Host Name** | **IP** | **Role** | **Remark** |
| cms.databox.com.tw | 192.168.6.100  192.168.5.60 | Cloudera Manager Server and Mangement service | 192.168.5.60唯對外IP，可連Internet |
| cm1.databox.com.tw | 192.168.6.101 | Management Node |  |
| cm2.databox.com.tw | 192.168.6.102 | work node |  |
| cm3.databox.com.tw | 192.168.6.103 | work node |  |
| cm4.databox.com.tw | 192.168.6.104 | work node |  |
| cm5.databox.com.tw | 192.168.6.105 | work node |  |

以cms.databox.com.tw為例 ，Server 主要配置如下:

1. **網路**

* sudo

因為不用root操作，所以新增一帳號且須有sudo權限

|  |
| --- |
| root#>groupadd bdmgr  root#>useradd -g bdmgr bdmgr  root#>passwd bdmgr  root#> sudo visudo  bdmgr ALL=(ALL) ALL  %bdmgr ALL=(ALL) NOPASSWD: ALL  #> su - bdmgr |

* DNS mapping

因為沒有建立DNS server，所以在/etc/hosts手動指定Nodes的FQDN name及對應的IP

#> sudo vi /etc/hosts

|  |
| --- |
| 192.168.6.100 cms.databox.com.tw cms  192.168.6.101 cm01.databox.com.tw cm01  192.168.6.102 cm02.databox.com.tw cm02  192.168.6.103 cm03.databox.com.tw cm03  192.168.6.104 cm04.databox.com.tw cm04  192.168.6.105 cm05.databox.com.tw cm05  192.168.6.109 cmbase.databox.com.tw cmbase |

#> sudo vi /etc/sysconfig/network

|  |
| --- |
| HOSTNAME=cms.databox.com.tw |

set host name permanently

|  |
| --- |
| #> hostnamectl set-hostname cms.databox.com.tw |

* network-adaptor setting

#> sudo vi /etc/sysconfig/network-scripts/ifcfg-eth0

|  |
| --- |
| BOOTPROTO=static  IPADDR=192.168.6.100  NETMASK=255.255.255.0  IPV6INIT=no  ONBOOT=yes |

1. **access without password**

|  |
| --- |
| ##generate public-key and private -key  #> ssh-keygen  ###copy public-key to remote-hosts  #> ssh-copy-id -i ~/.ssh/id\_rsa.pub remote-hosts |

1. **防火牆**

* SELINUX關閉

|  |
| --- |
| #> sudo setenforce 0  #> sudo vi /etc/sysconfig/selinux  SELINUX=disabled |

* firewalld關閉

|  |
| --- |
| #> sudo systemctl disable firewalld  #> sudo systemctl stop firewalld |

1. **mount iso file to local repository**

|  |
| --- |
| 1. copy iso file to /opt/yumrepos/centosiso 2. mount iso file   #> mkdir /tmp/isorepo  #> sudo mount -o loop /opt/yumrepos/centosiso/CentOS-7-x86\_64-DVD-1511.iso /tmp/isorepo   1. create local iso repository   #> sudo vi /etc/yum.repos.d/iso.repo  [iso-repository]  name=iso-repository  baseurl=file:///tmp/isorepo/  enabled=1  gpgcheck=0  #gpgcheck=1  #gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release   1. disable redhat.repo   #> vi /etc/yum.reposd/redhat.repo  [rhel-6-server-rpms] ===> enabled = 0  #> sudo mv /etc/yum.reposd/redhat.repo /etc/yum.reposd/redhat.repo.disabled   1. clean cache   #> sudo rm -rf /var/cache/yum/x86\_64/  #> sudo yum clean all   1. 如需要永久mount此iso file則可參考如下步驟   #> sudo vi /etc/fstab  /tmp/rhel-server-6.7-x86\_64-dvd.iso /tmp/isorepo iso9660 loop 0 0 |

1. **安裝基本套件**

|  |
| --- |
| #> sudo yum install bind-utils |
|  |

1. **NTP設定**

|  |
| --- |
| #> sudo yum install ntp  #> sudo systemctl start ntpd  #> sudo systemctl enable ntpd |

Local ntp server setup

|  |
| --- |
| #> sudo vi /etc/ntp.conf  restrict 192.168.6.0 mask 255.255.255.0 nomodify notrap  #> sudo systemctl restart ntpd  ##必須至少要可以連到下列清單的其中一台，所以gateway必須打開  server 0.rhel.pool.ntp.org iburst  server 1.rhel.pool.ntp.org iburst  server 2.rhel.pool.ntp.org iburst  server 3.rhel.pool.ntp.org iburst  #> sudo ntpdate -u 0.rhel.pool.ntp.org  #> sudo hwclock --systohc |

Local ntp client setup

|  |
| --- |
| #> sudo vi /etc/ntp.conf  server cms.databox.com.tw  #> sudo service ntpd restart  #> ntpstat |

1. **swapiness=0**

|  |
| --- |
| #> cat /proc/sys/vm/swappiness  #> sudo sysctl vm.swappiness=0  #> sudo vi /etc/sysctl.conf  vm.swappiness=0 |

1. **disable ipv6**

|  |
| --- |
| #> sudo vi /etc/sysctl.conf  net.ipv6.conf.all.disable\_ipv6=1  net.ipv6.conf.default.disable\_ipv6=1  net.ipv6.conf.lo.disable\_ipv6=1 |

1. **Setting the Ulimits**

|  |
| --- |
| #> sudo vi /etc/security/limits.conf  \* hard nproc 32768  \* soft nproc 32768  \* hard nofile 32768  \* soft nofile 32768 |

1. **vi /etc/rc.local**

|  |
| --- |
| #> sudo vi /etc/rc.local  echo never > /sys/kernel/mm/transparent\_hugepage/defrag |

1. **copy jdbc jar file to /usr/share/java**

|  |
| --- |
| #> sudo mkdir /usr/share/java  #> cd /usr/share/java  #> sudo wget <http://central.maven.org/maven2/mysql/mysql-connector-java/5.1.40/mysql-connector-java-5.1.40.jar>  #> sudo mv mysql-connector-java-5.1.40.jar mysql-connector-java.jar |

1. **cms.databox.com.tw專屬設定**

因為cms.databox.com.tw須有對外網路，所以多了一組network-adaptor

#> sudo vi /etc/sysconfig/network-scripts/ifcfg-eth1

|  |
| --- |
| BOOTPROTO=static  IPADDR=192.168.5.61  NETMASK=255.255.255.0  GATEWAY=192.168.5.168  DNS1=8.8.8.8  IPV6INIT=no  ONBOOT=yes |

安裝local repository 及安裝 httpd

|  |
| --- |
| #> sudo yum install createrepo  #> sudo yum install httpd  #> sudo systemctl start httpd  #> sudo systemctl enable httpd  #> sudo mkdir /var/www/html/repo  #> sudo ln -s /tmp/isorepo/ /var/www/html/repo/isoRepo  #> sudo vi /etc/yum.repos.d/iso.repo  baseurl=<http://cms.databox.com.tw/repo/isoRepo>  ##開啟Browser並且測試[http://192.168.5.60/repo/](http://192.168.5.61/repo/)isoRepo |

1. **Cloudera Manager Software download**

在cms.databox.com.tw node下載所需的元件

1. **download CM component**

|  |
| --- |
| #> mkdir /opt/yumrepos/cm  #> cd /opt/yumrepos/cm  #> wget -c -r -nd -np -K -L -A rpm http://archive-primary.cloudera.com/cm5/redhat/7/x86\_64/cm/5.8.3/RPMS/x86\_64/  #> wget http://archive-primary.cloudera.com/cm5/redhat/7/x86\_64/cm/RPM-GPG-KEY-cloudera |

1. **download Parcels**

|  |
| --- |
| #> sudo mkdir /opt/yumrepos/parcels  #> cd /opt/yumrepos/parcels  #> wget http://archive-primary.cloudera.com/cdh5/parcels/5.8.3/CDH-5.8.3-1.cdh5.8.3.p0.2-el7.parcel  #> wget <http://archive-primary.cloudera.com/cdh5/parcels/5.8.3/manifest.json> |

1. **create local repository**

|  |
| --- |
| #> cd /opt/yumrepos/cm  #> sudo createrepo .  #> sudo rm -rf /var/cache/yum/x86\_64/6Server/  #> sudo yum clean all  #> sudo ln -s /opt/yumrepos/cm/ /var/www/html/repo/cm |

1. **test Local repository**

開啟Browser並且測試[http://192.168.5.60/repo/](http://192.168.5.61/repo/)cm/

1. **create myrepo.repo**

建立新檔案至/etc/yum.reposd/myrepo.repo

|  |
| --- |
| #> sudo chmod +w /etc/yum.repos.d/  #> sudo vi /etc/yum.repos.d/cm.repo  [cm]  name=cm  baseurl=<http://cms.databox.com.tw/repo/cm/>  enables=true  gpgcheck=false |

1. **Install Cloudera Manger**
2. **install JDK**

|  |
| --- |
| #> sudo yum install oracle-j2sdk1.7  #> vi ~/.bash\_profile  export JAVA\_HOME=/usr/java/jdk1.7.0\_67-cloudera/  export JRE\_HOME=/usr/java/jdk1.7.0\_67-cloudera/jre  PATH=$JAVA\_HOME/bin:$PATH:$HOME/bin  #> source ~/.bash\_profile |

安裝SDK動作也可在其他node先執行

1. **install Cloudera Mangager**

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

1. **install mySQL DB**

此步驟離線上安裝方式，若可以線上安裝，請參考附註一

|  |
| --- |
| #> cd /tmp  #> sudo mkdir /tmp/mysql  #> sudo wget <http://dev.mysql.com/get/Downloads/MySQL-5.7/mysql-5.7.17-1.el7.x86_64.rpm-bundle.tar>  #> sudo tar -C /tmp/mysql -xvf mysql-5.7.17-1.el7.x86\_64.rpm-bundle.tar  #/> cd /tmp/mysql  #> sudo yum install mysql-community-{server,client,common,libs}-\*  在my.cnf加入下方參數  #> sudo vi /etc/my.cnf  transaction-isolation = READ-COMMITTED  max\_connections = 150  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  啟動service  #> sudo systemctl start mysqld  ##從/var/log/mysqld.log找出預設的密碼  #> sudo grep 'temporary password' /var/log/mysqld.log  ##root密碼改為P@ssw0rd  #> sudo mysql\_secure\_installation  #> sudo /sbin/chkconfig mysqld on |

1. **[參考章節]建議將db data搬移至某一獨立磁碟，以避免root folder容量不足問題發生**

|  |
| --- |
| #> sudo systemctl stop mysqld  #> su - root  #> cd /var/lib/  #> mv mysql /data1/mysql  #> mv mysql-files /data1/mysql-files  #> mv mysql-keyring /data1/mysql-keyring  #> ln -s /data1/mysql mysql  #> chown -h mysql:mysql mysql  #> ln -s /data1/mysql-files mysql-files  #> chown -h mysql:mysql mysql-files  #> ln -s /data1/mysql-keyring mysql-keyring  #> chown -h mysql:mysql mysql-keyring  #> exit  #> sudo systemctl start mysqld |

1. **start mySQL DB**

|  |
| --- |
| #> sudo mysql -uroot -p  mysql> create database cmdb DEFAULT CHARACTER SET utf8;  mysql> grant all on cmdb.\* TO 'cmuser'@'%' IDENTIFIED BY 'P@ssw0rd';  mysql> create database hivedb DEFAULT CHARACTER SET utf8;  mysql> grant all on hivedb.\* TO 'hiveuser'@'%' IDENTIFIED BY 'P@ssw0rd';  mysql> create database rptmgrdb DEFAULT CHARACTER SET utf8;  mysql> grant all on rptmgrdb.\* TO 'rptmgruser'@'%' IDENTIFIED BY 'P@ssw0rd';  mysql> create database ooziedb DEFAULT CHARACTER SET utf8;  mysql> grant all on ooziedb.\* TO 'oozieuser'@'%' IDENTIFIED BY 'P@ssw0rd';  mysql> create database huedb DEFAULT CHARACTER SET utf8;  mysql> grant all on huedb.\* TO 'hueuser'@'%' IDENTIFIED BY 'P@ssw0rd';  mysql>exit; |

1. **產生db properties檔給ClouderaManager用**

|  |
| --- |
| #> sudo /usr/share/cmf/schema/scm\_prepare\_database.sh mysql cmdb cmuser P@ssw0rd |

1. **start Cloudera Manager server**

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

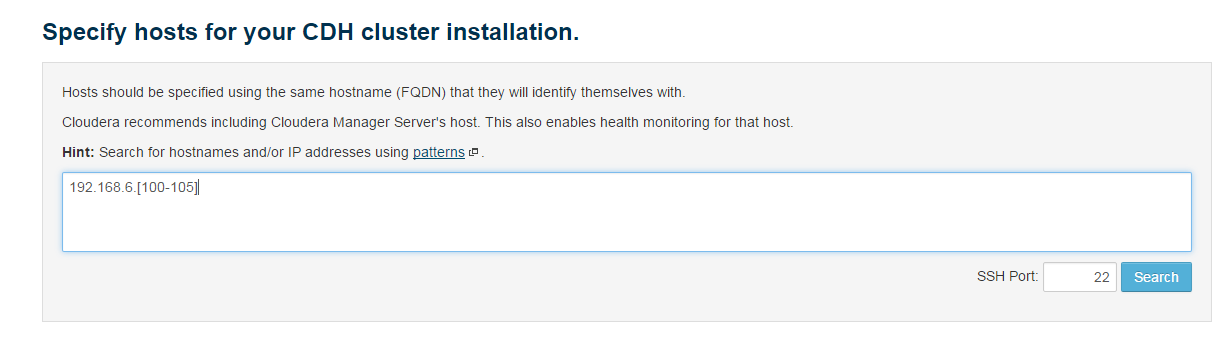
1. **Install Hadoop CDH**

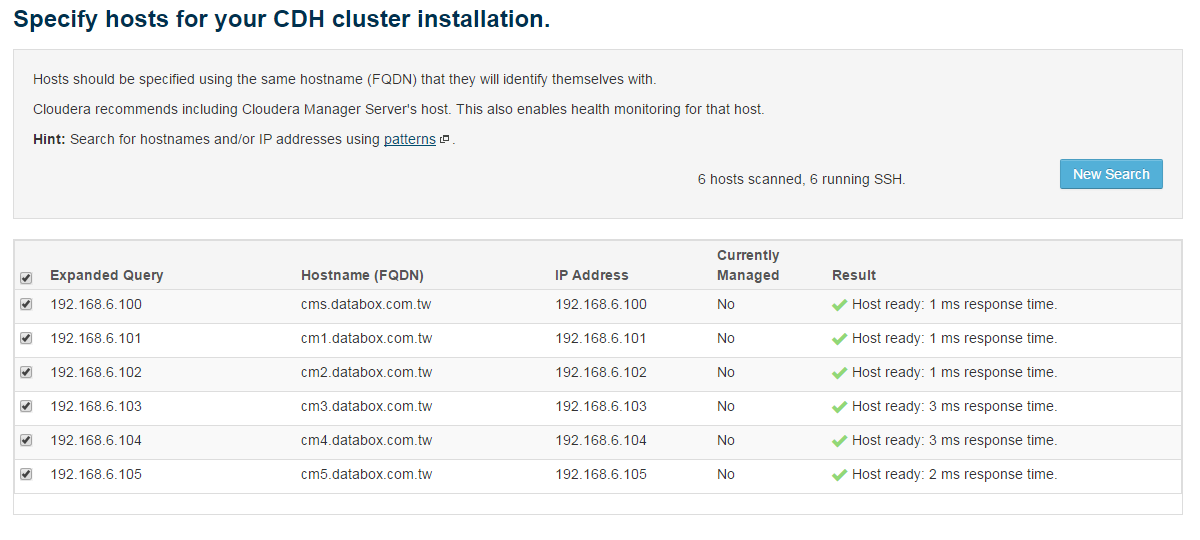
透過Cloudera Manager來安裝Hadoop CDH5.8.3

1. **sarch hosts**

找出所有的host，可用IP區間或是host name

|  |
| --- |
| 192.168.6.[100-105] |





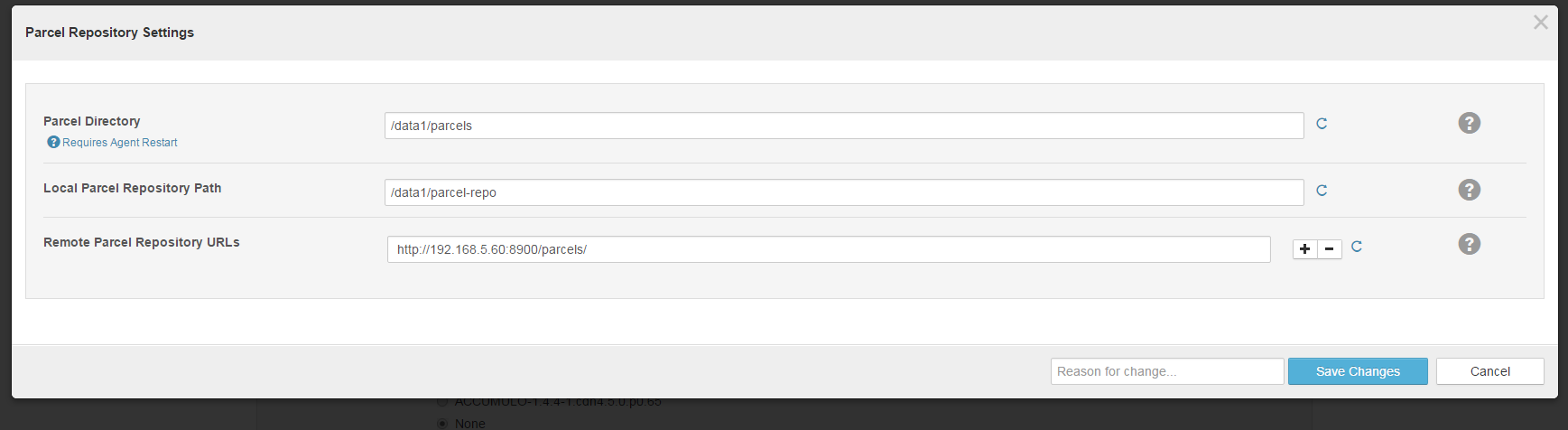
1. **設定Repository and Parcels**

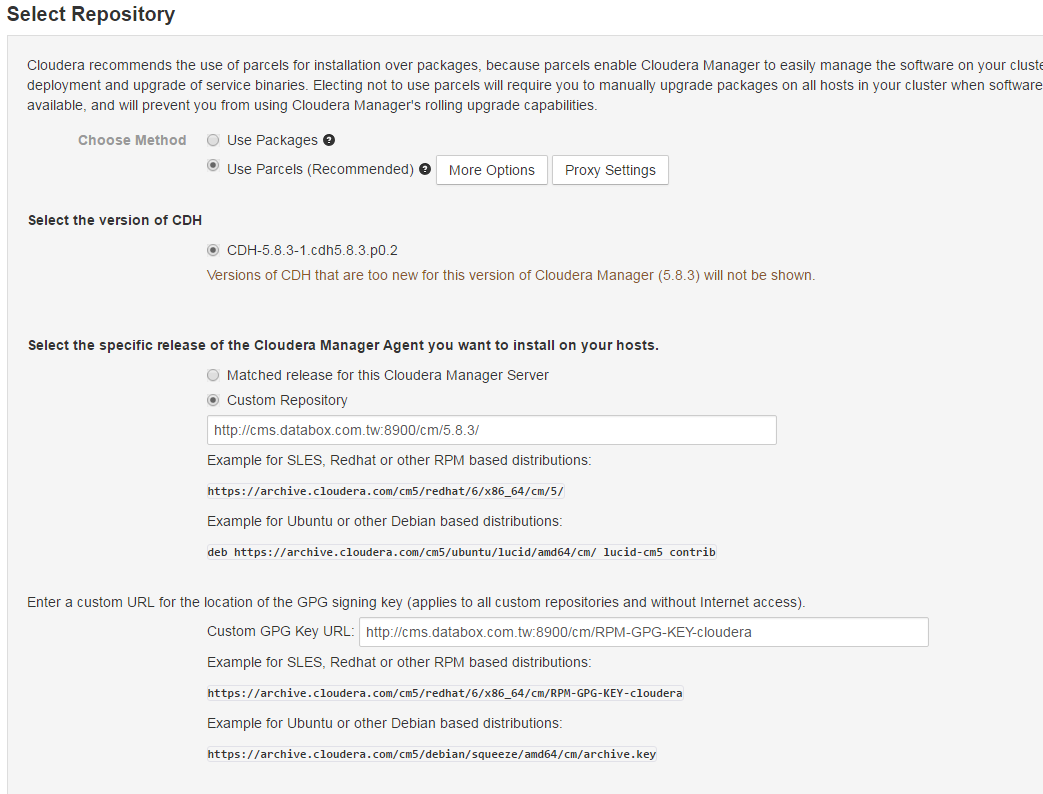
**Parcel Repository Settings**

|  |  |  |
| --- | --- | --- |
| **Key** | **Value** | **Remark** |
| **Parcel Directory** | /opt/cloudera/parcels |  |
| **Local Parcel Repository Path** | /opt/cloudera/parcel-repo |  |
| **Remote Parcel Repository URLs** | http://192.168.5.60/repo/parcels/ |  |

**Select the specific release of the Cloudera Manager Agent you want to install on your hosts**

|  |  |  |
| --- | --- | --- |
| **Key** | **Value** | **Remark** |
| Custom Repository | http://cms.databox.com.tw/repo/cm/ |  |
| Custom GPG Key URL | http://cms.databox.com.tw/repo/cm/RPM-GPG-KEY-cloudera |  |





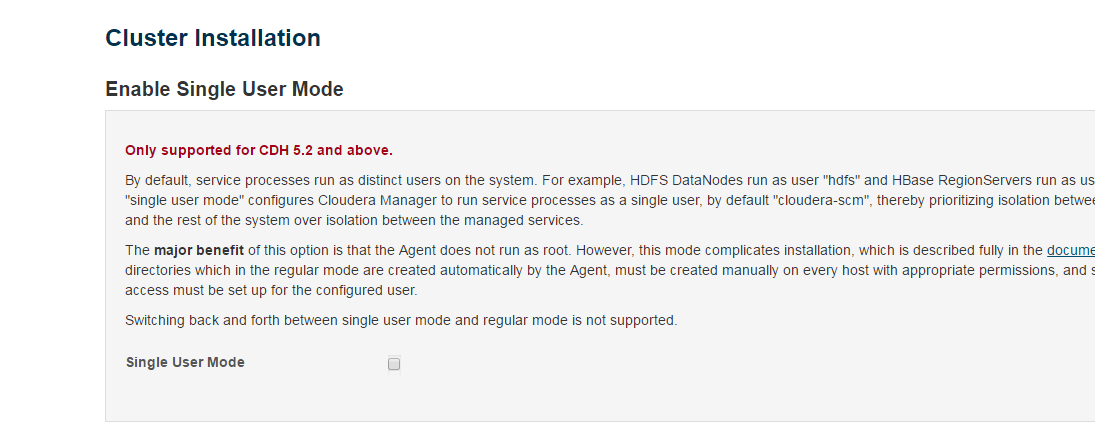
1. **Install JDK in Hosts**

若已經安裝JDK，則可以略過此步驟

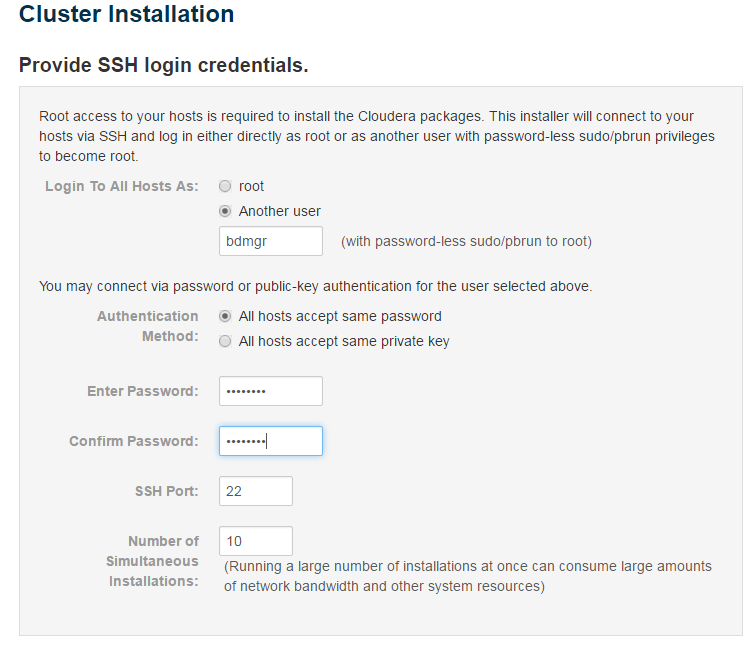


1. **Single User Mode**

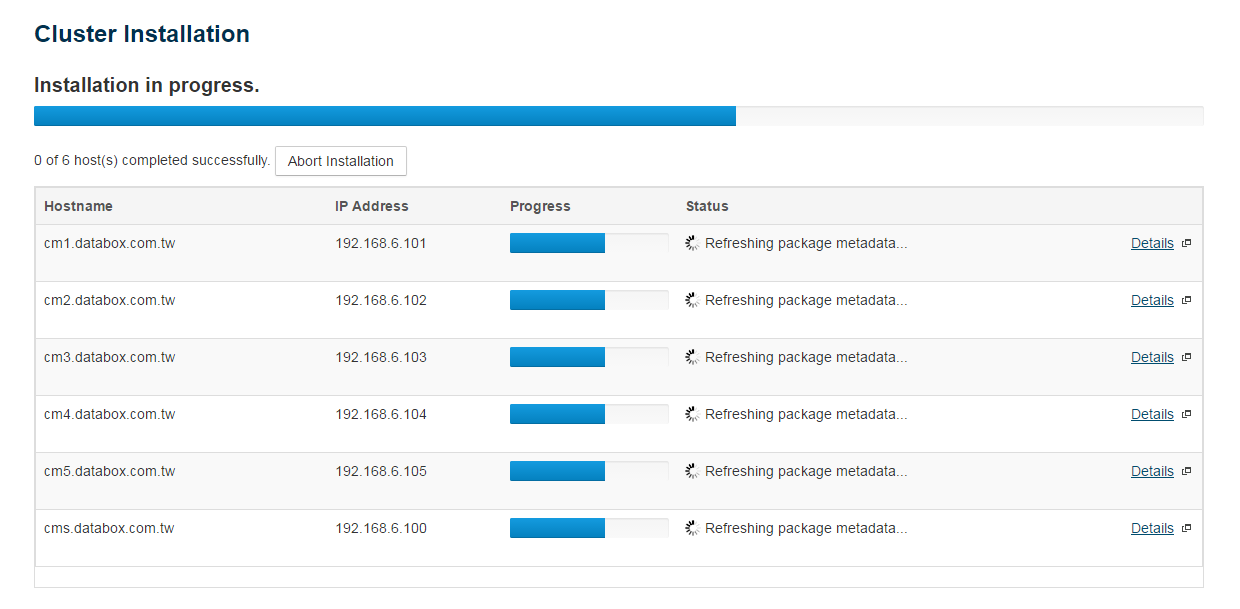
略過此項，**之前勾選此項都會遇到問題**

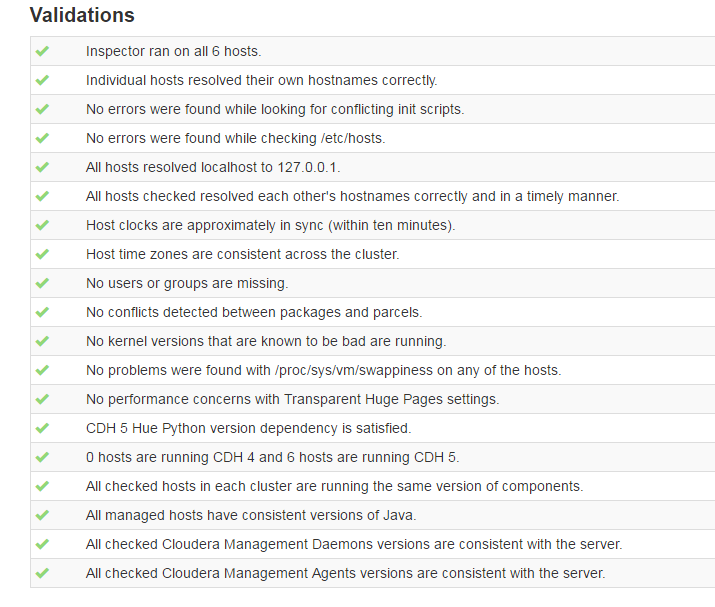


1. **Agent user**



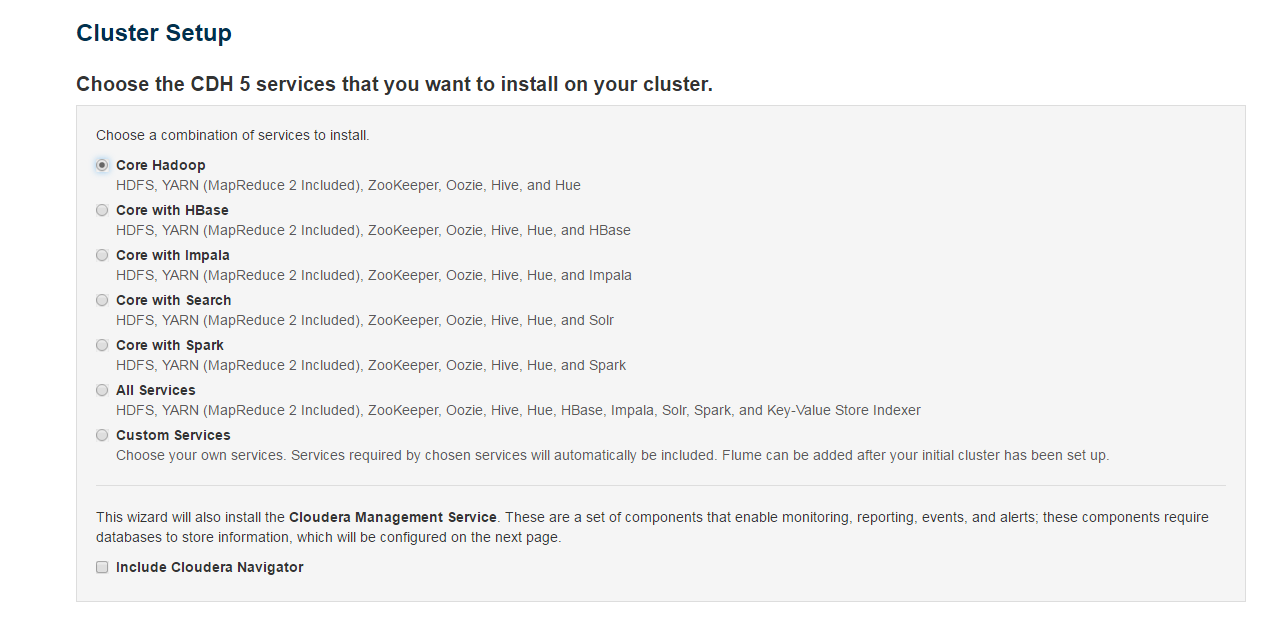
1. **Agent Install**





1. **Cluster Configuration Setup**

選擇基本安裝

****

**以下為設定每台機器的角色，主要規劃會分為下列幾大類**

1. **Management Node**: 主要是NameNode, SecondaryNameNode, ResourceManager, ZooKeeper等
2. **Work Node**: Data Node, NodeManager, RegionServer
3. **Infrastructure Node**: DNS, CM, Hue, Hive
4. **Edge Node**: 同時有外部IP及內部IP，對外聯結的跳板機

* HDFS

|  |  |  |
| --- | --- | --- |
| Key | Value | Type |
| NameNode | cm1 | Management Node |
| SecondaryNameNode | cm2 | Management Node |
| Balancer | cm2 | Management Node |
| HttpFS | NA | Management Node |
| NFS Gateway | NA |  |
| DataNode | cm[2-5], cms | Work Node |

* Hive

|  |  |  |
| --- | --- | --- |
| Key | Value | Type |
| Gateway | cm[1-5], cms |  |
| Hive Metastore Sever | cm3 | Infrastructure Node |
| WebHCat Server | cm3 | Infrastructure Node |
| HiveServer2 | cm3 | Infrastructure Node |

* Hue

|  |  |  |
| --- | --- | --- |
| Key | Value | Type |
| Hue Server | cm3 | Infrastructure Node |

* Cloudera Management Service

將ClouderaManagement Services全改為cms.databox.com.tw

|  |  |  |
| --- | --- | --- |
| Key | Value | Type |
| Service monitor | cms | Infrastructure Node |
| Activity Monitor | NA |  |
| Host Monitor | cms | Infrastructure Node |
| Reports Manager | cms | Infrastructure Node |
| Event Server | cms | Infrastructure Node |
| Alert Publisher | cms | Infrastructure Node |

* Oozie

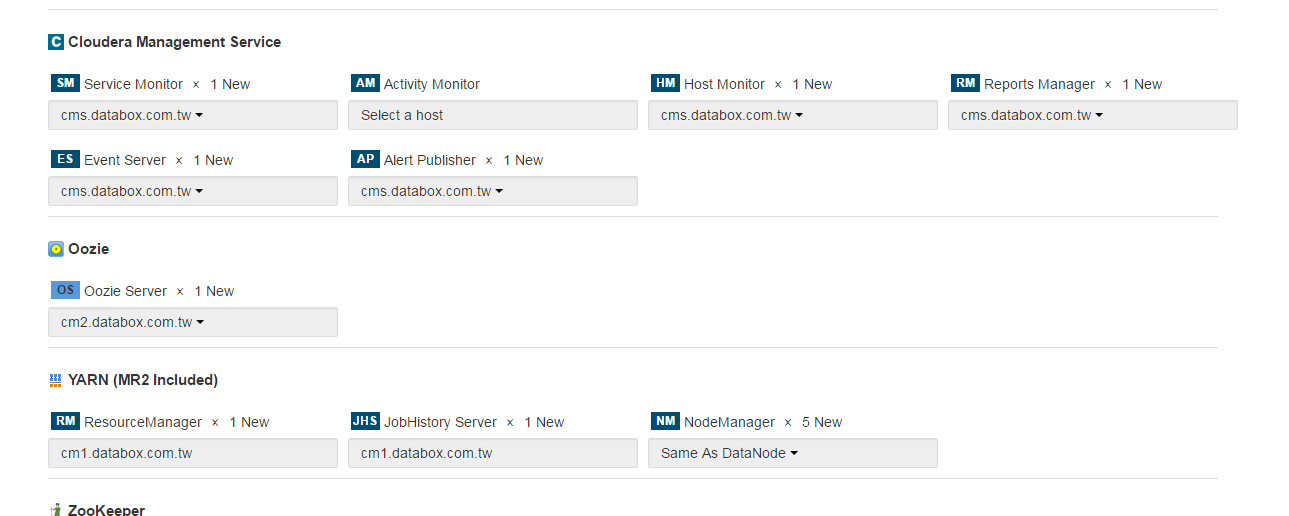
|  |  |  |
| --- | --- | --- |
| Key | Value | Type |
| Oozie Server | cm3 | Infrastructure Node |

* YARN

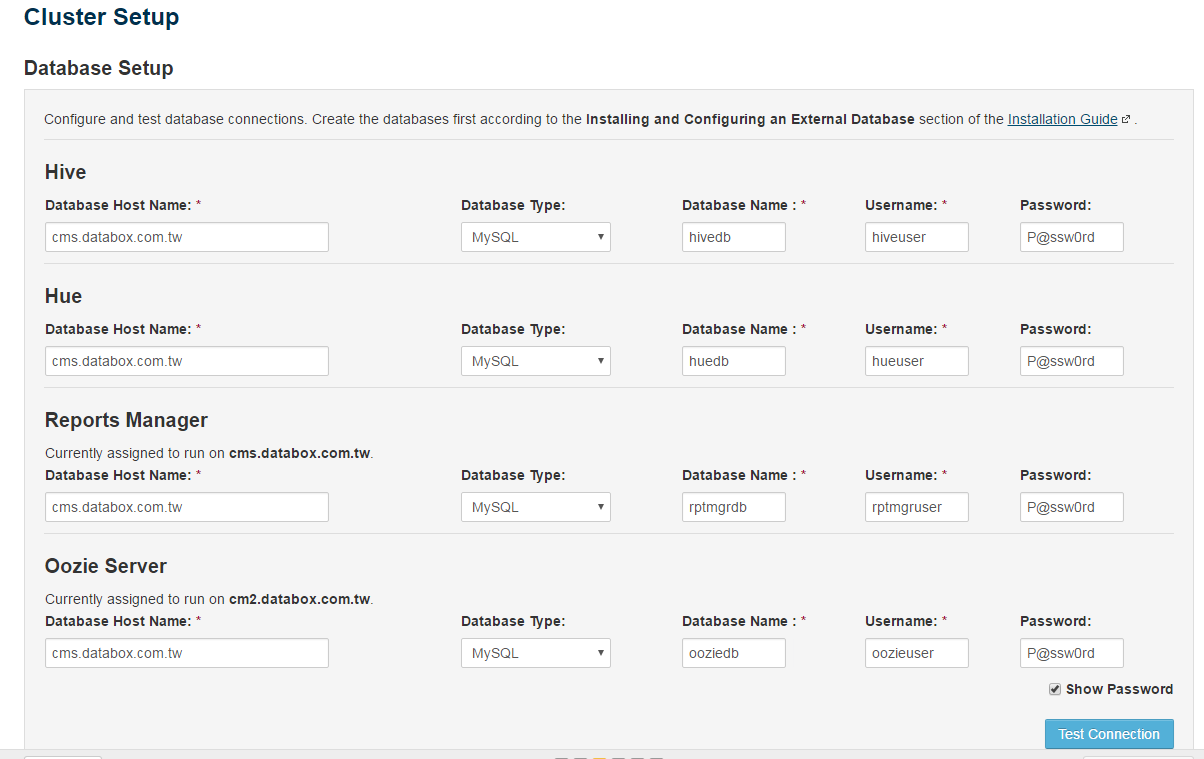
|  |  |  |
| --- | --- | --- |
| Key | Value | Type |
| ResourceManager | cm1 | Management Node |
| JobHistory Server | cm1 | Management Node |
| NodeManager | cm[2-5], cms | Same as DataNode |

* ZooKeeper

|  |  |  |
| --- | --- | --- |
| Key | Value | Type |
| Server | cm1 | Management Node |

****

輸入各元件所需的RDB設定

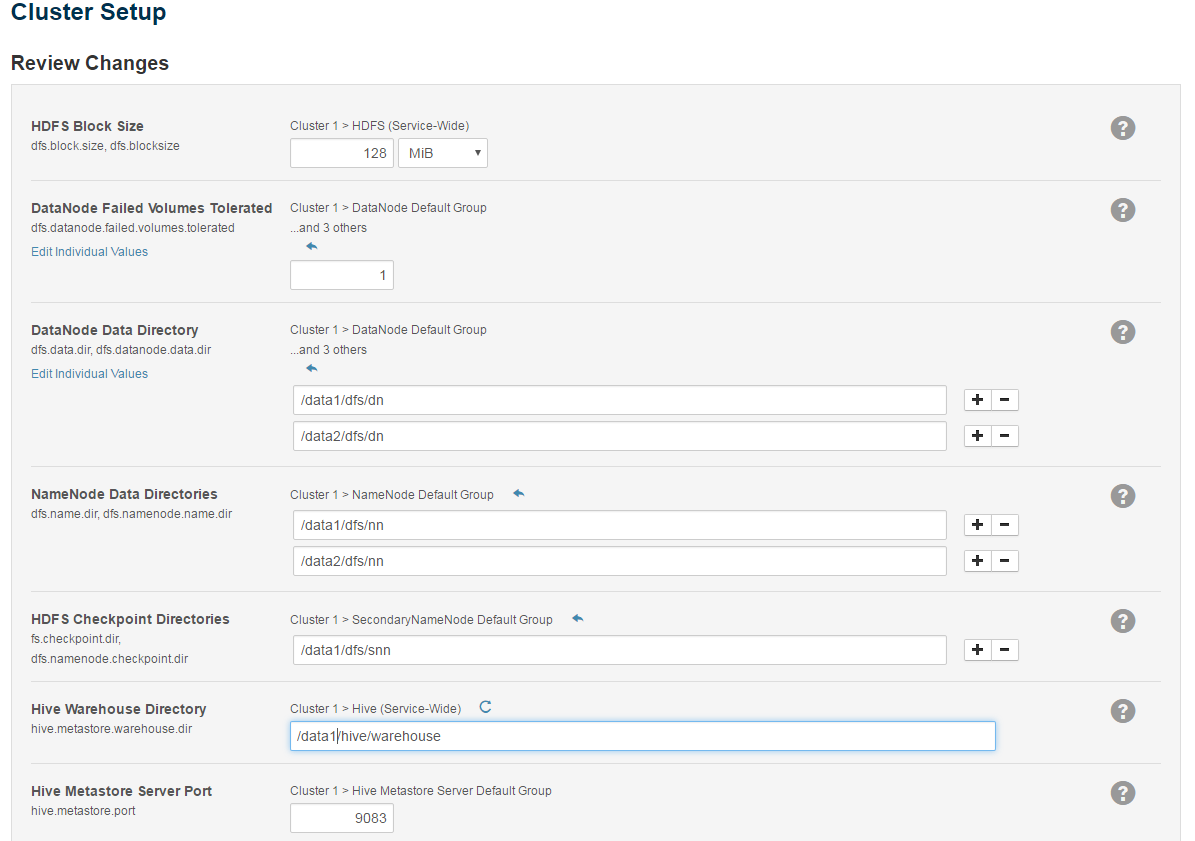
****

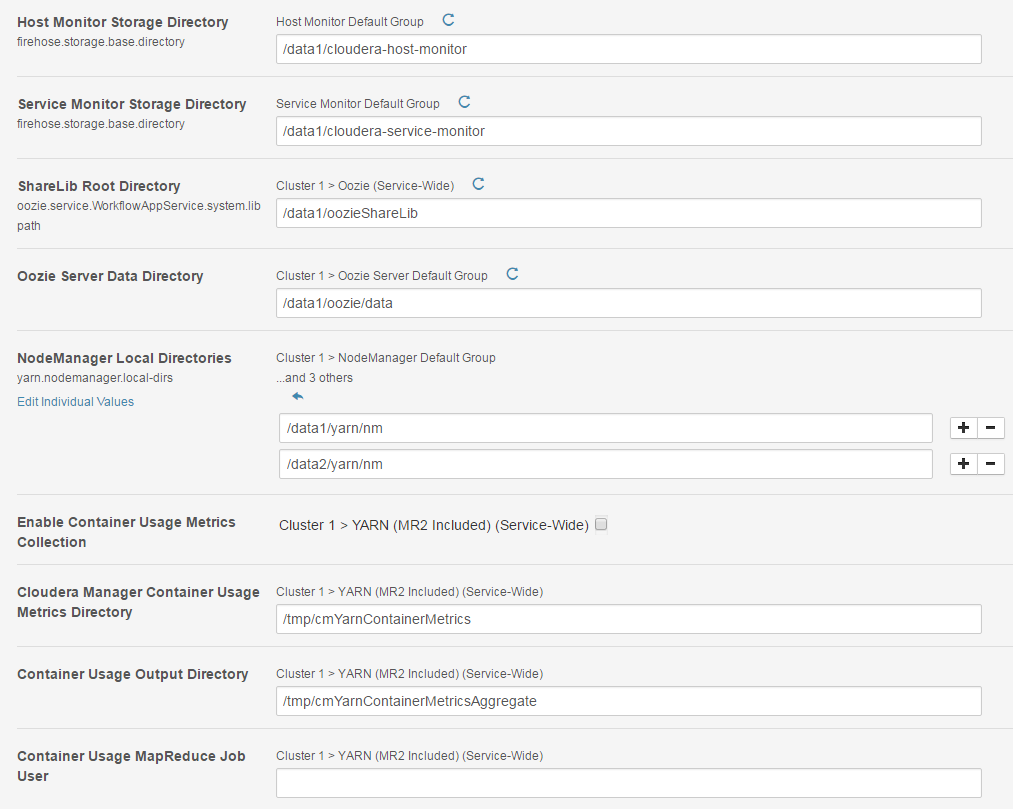
**CluserSetup**

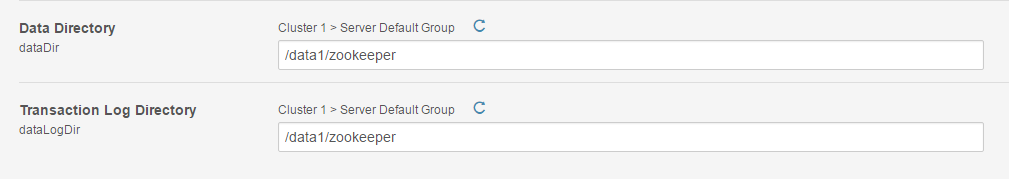
**目前測試的VM有三顆硬碟，第一顆為Linux OS使用，第二顆monunt為/data1，第三顆monunt為/data2**

* Data Node Directory

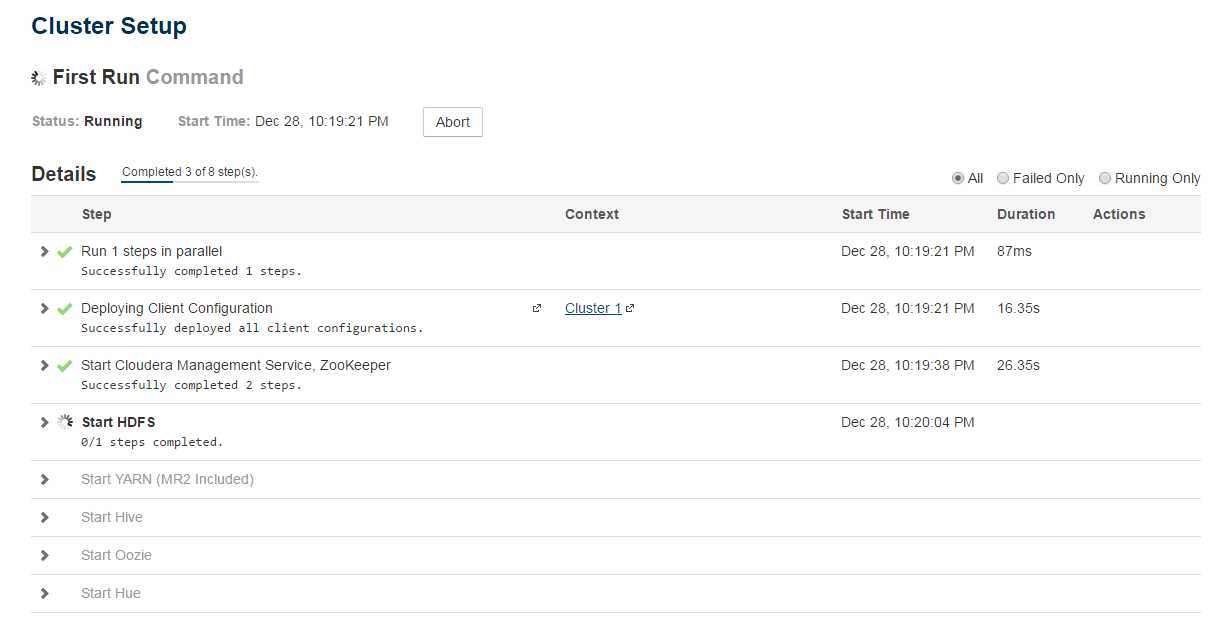
|  |  |  |
| --- | --- | --- |
| Key | Value | Remark |
| Data Node Directory | /data1/dfs/dn, /data2/dfs/dn |  |
| NameNode Directory | /data1/dfs/nn, /data2/dfs/nn |  |
| HDFS Checkpoint Directory | /data1/dfs/snn |  |

****

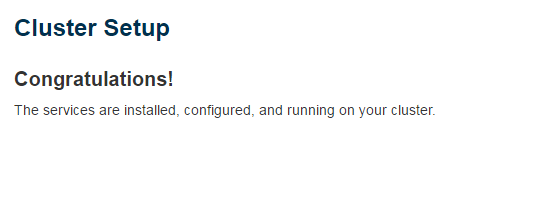
****

****

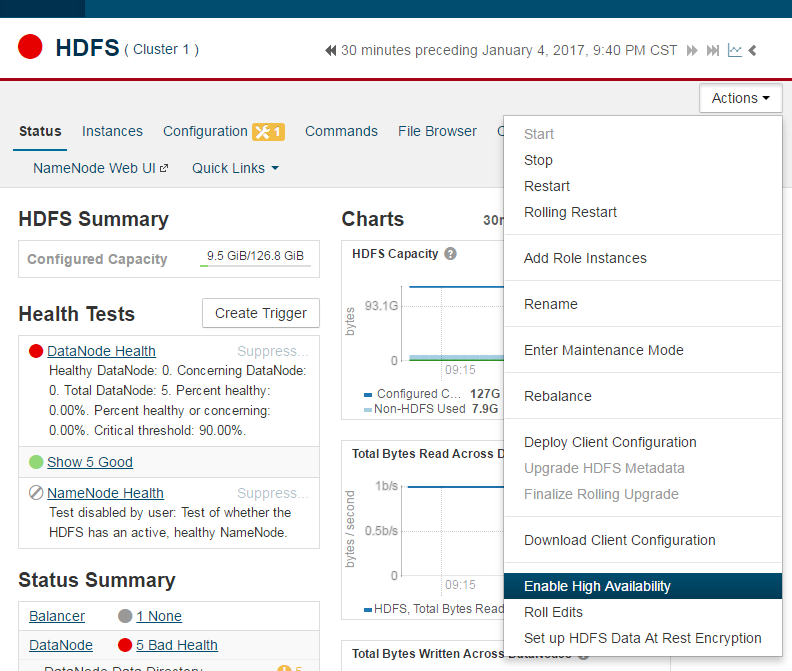
設定完成，開始deployment

****

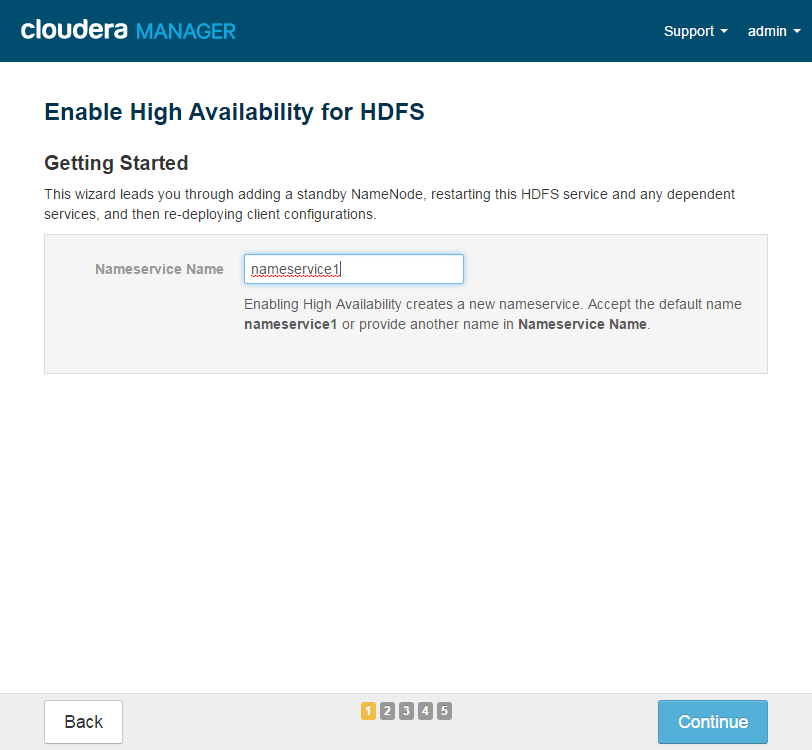
安裝完成



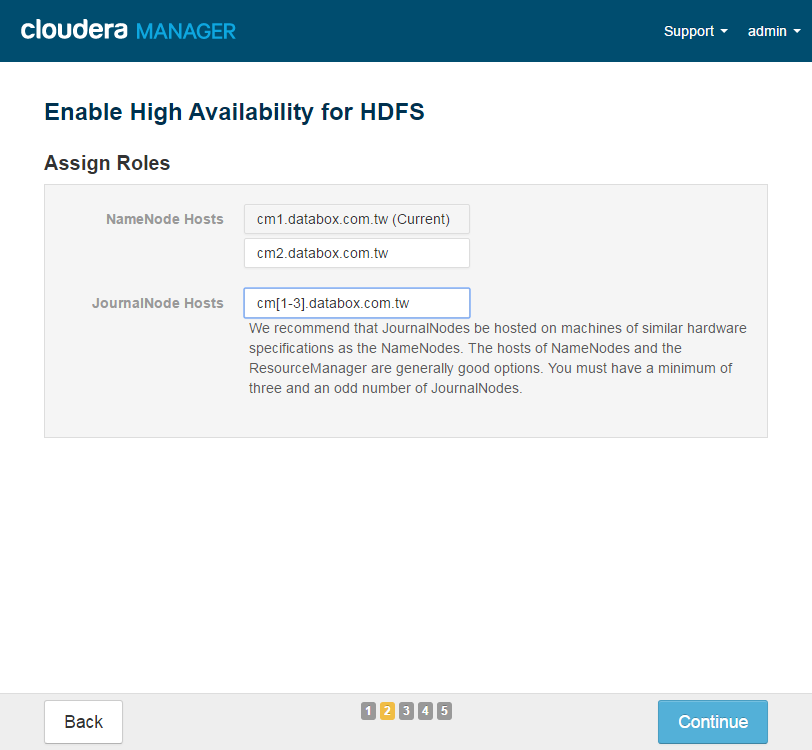
1. **Enable High Availability**
2. **Enable HDFS HA**

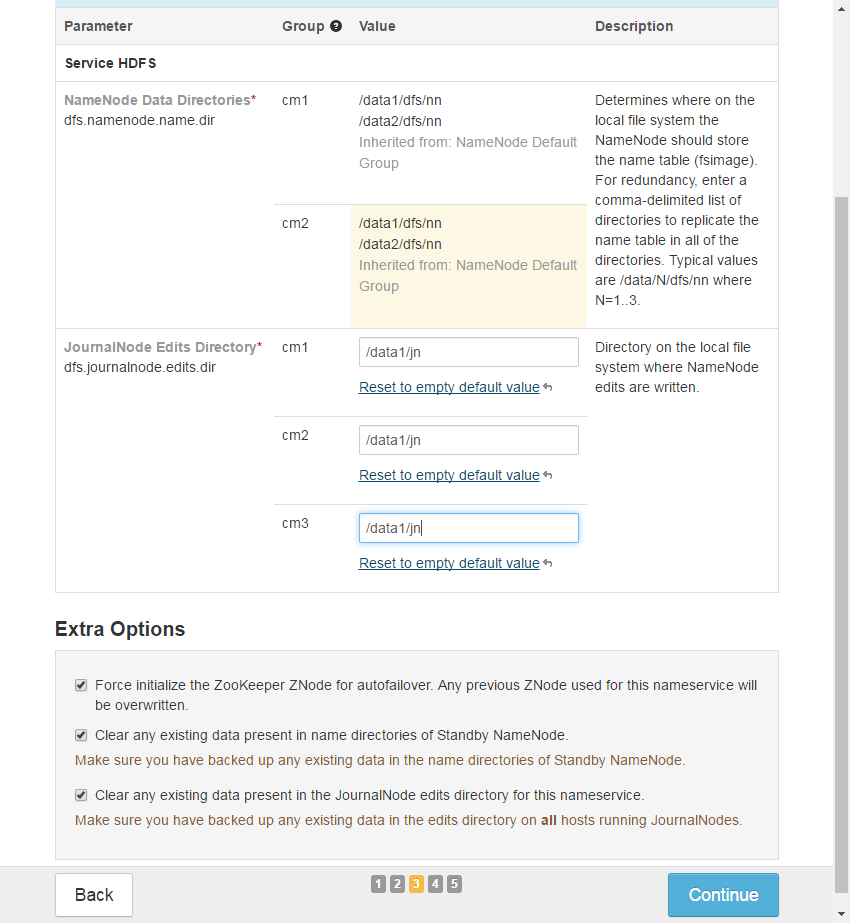
****

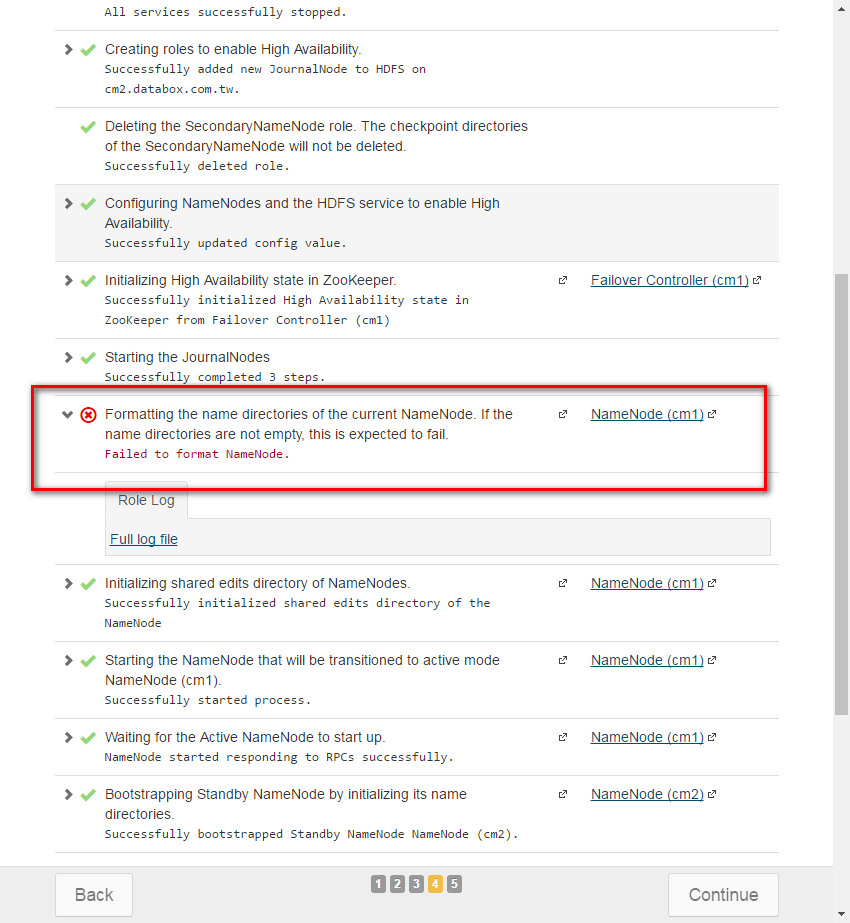
**請問這個NameService Name是真的是一個DNS Server還是只是一個內部使用的一個專屬nameService的Name**

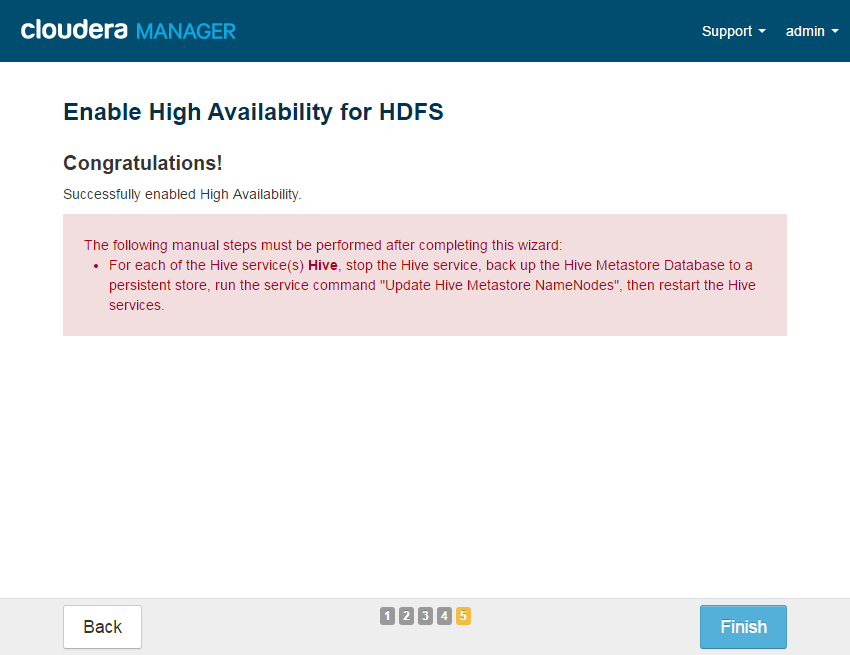
****

JoumalNode的host數量必須為奇數

****

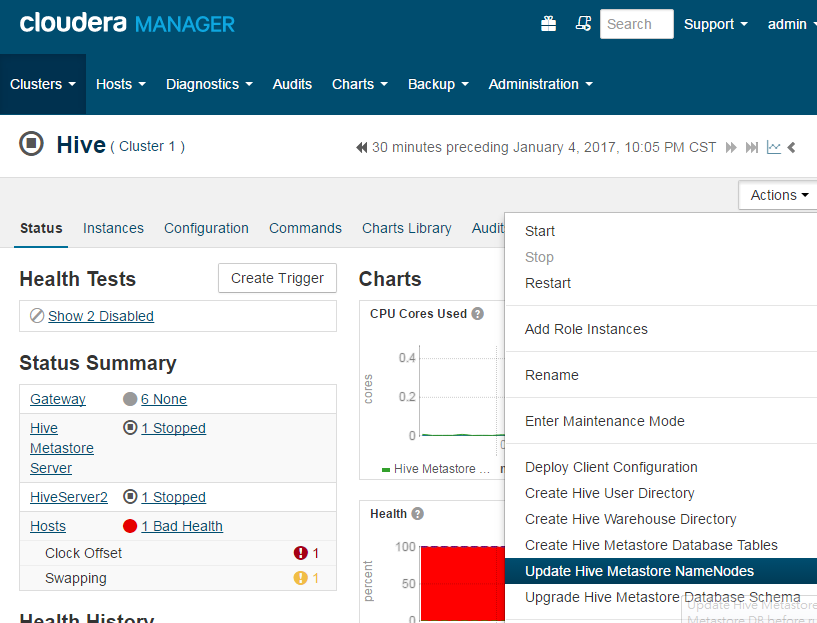
****

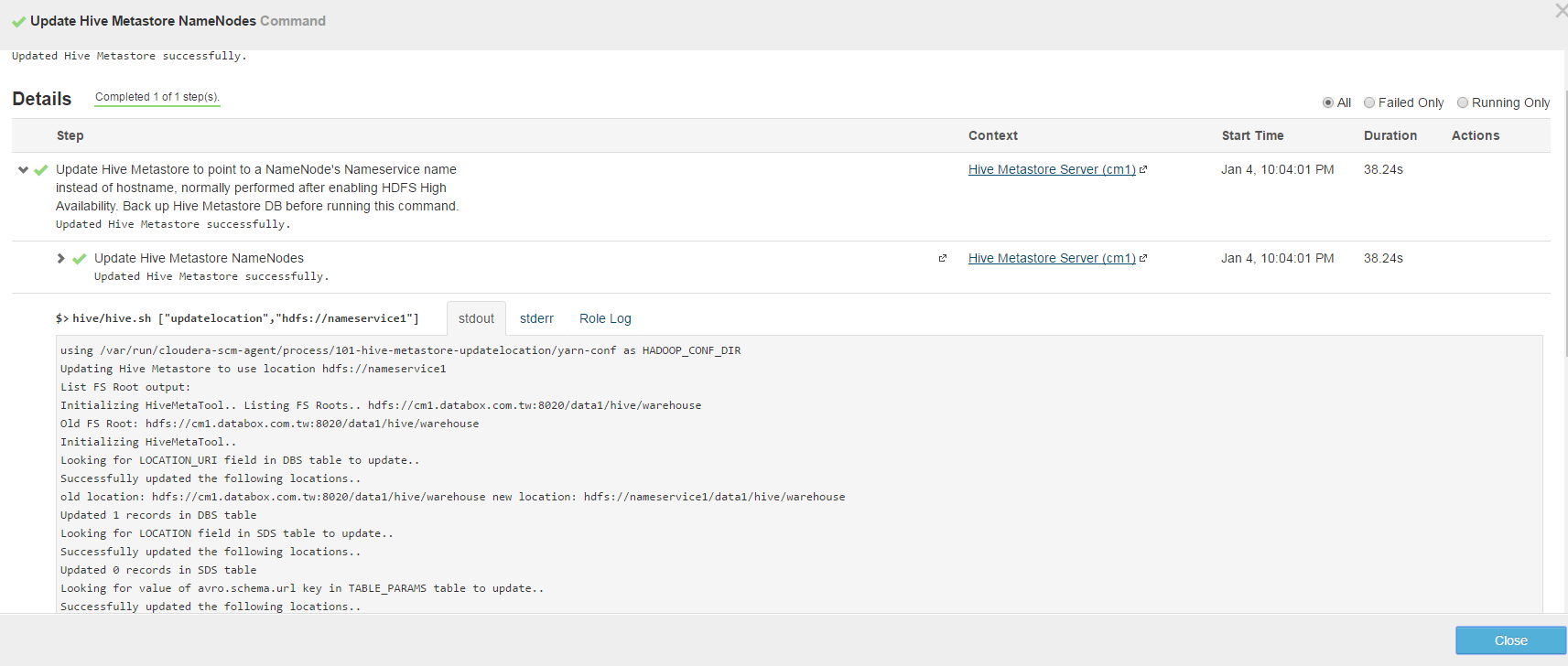
****

****

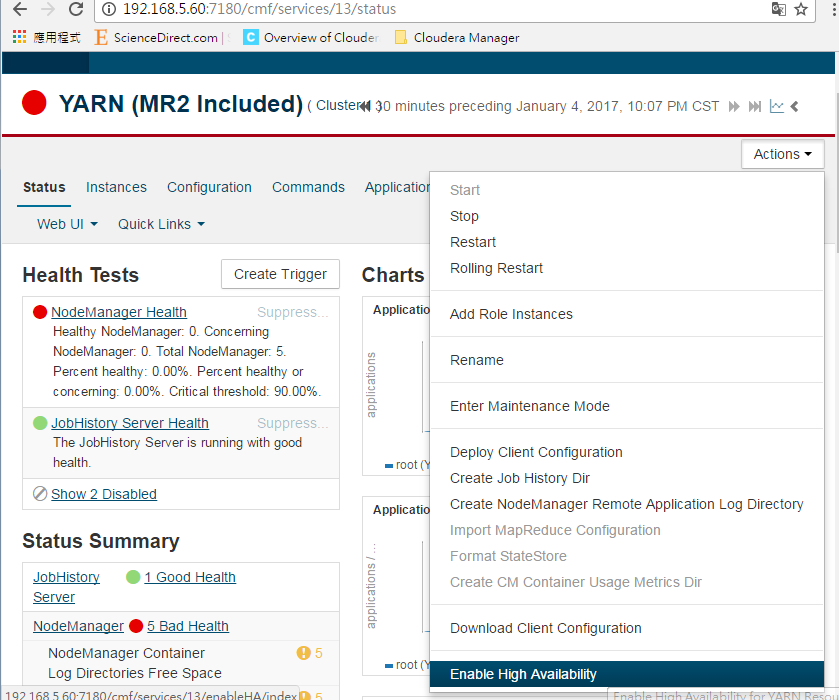
透過GUI設定完HDFS HA之後，必須再手動執行Hive相關操作：

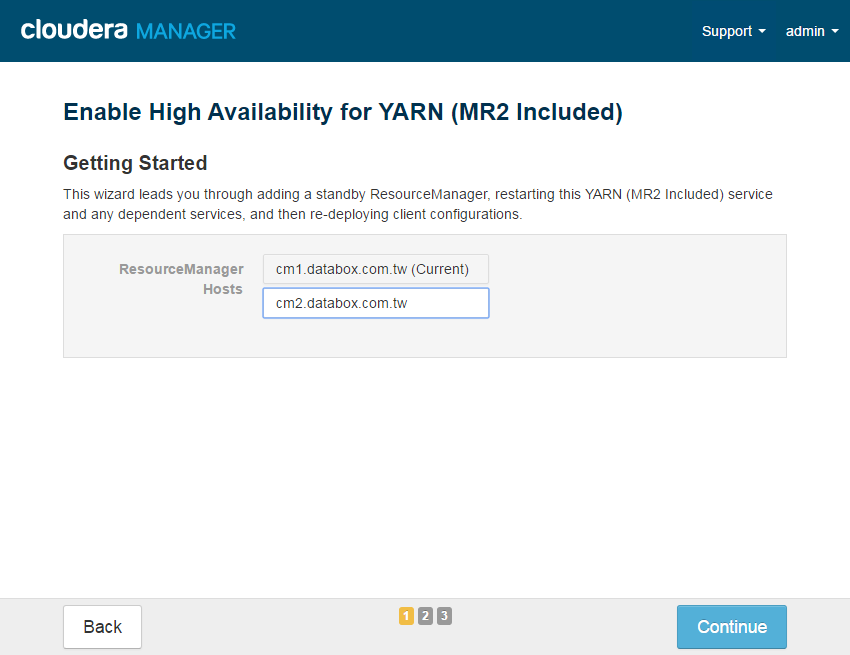
1. Stop Hive service
2. 執行 'Update Hive Metastore NameNodes’

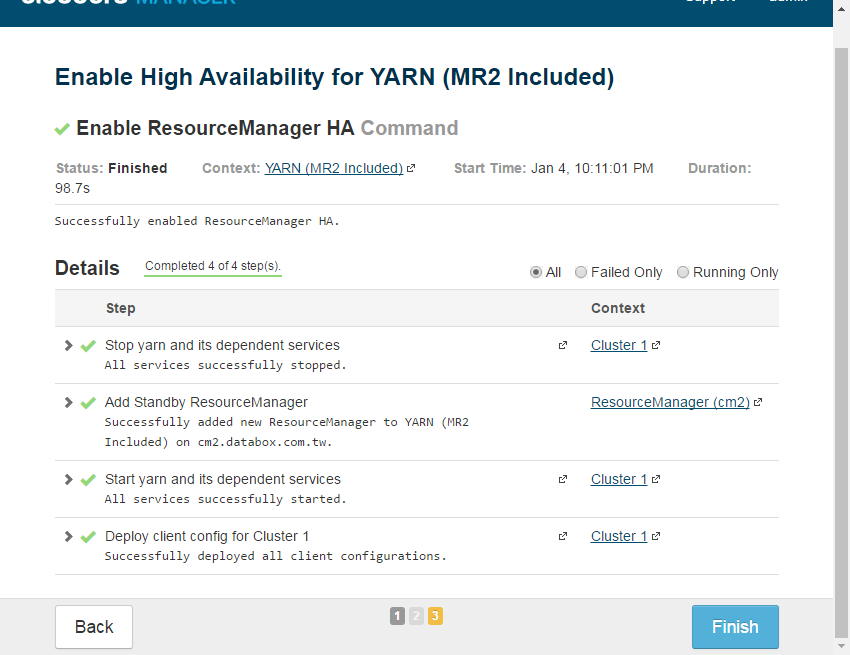
****

****

1. **Enable YARN HA**

****

****

****

1. **Installed Validation**
2. 安裝後驗證步驟概述：
3. 執行一wordCount的MapReduce程式，執行前會先在HDFS建立測試資料，用來驗證HDFS及YARN元件應正常。
4. Create a Hive table，並且執行Hive Sql，用來驗證Hive、HDFS及YARN元件是否正常
5. 驗證NameNode HA方式：關閉Primary NameNode，執行wordCount and Hive SQL 應得到程式預期結果。
6. 驗證ResourceManager HA方式：關閉Primary ResourceManager，執行wordCount and Hive SQL 應得到程式預期結果。
7. 執行ozzie-example程式，用來驗證oozie元件是否正常
8. 執行MapReduce程式

|  |
| --- |
| #> sudo groupadd cloudera-dev  #> sudo useradd -g cloudera-dev cloudera-dev  #> sudo -u hdfs hadoop fs -mkdir /user/cloudera-dev  #> sudo -u hdfs hadoop fs -chown cloudera-dev:cloudera-dev /user/cloudera-dev  #> sudo su cloudera-dev  #> echo “Hello world, Bye world” > file0  #> echo “Hello Hadoop, Bye Hadoop” > file1  #> hadoop fs -mkdir -p /user/cloudera-dev/wordcount/input  #> hadoop fs -put file\* /user/cloudera-dev/wordcount/input  #> yarn jar /opt/cloudera/parcels/CDH/jars/hadoop-examples.jar wordcount wordcount/input wordcount/output  #> hadoop fs -getmerge wordcount/output output.txt  #> cat output.txt |

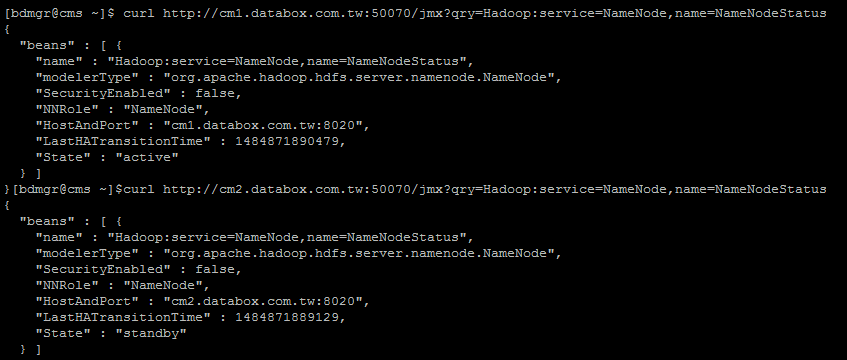
1. 執行Hive程式

|  |
| --- |
| #> sudo su - cloudera-dev  #> echo “Alex, Cloudrea” > file2  #> hadoop fs -mkdir -p hive/input  #> hadoop fs -put file2 /user/cloudera-dev/hive/input  #> vi test.hql  create external table test (  name String,  company String  )  row format delimited fields terminated by ','  location '/user/cloudera-dev/hive/input'  #> hive -f test.hql  #> hive -e "select \* from test" 2>/dev/null  #> hive -e "select count(\*) from test" 2>/dev/null  #> hive -e "drop table test" |

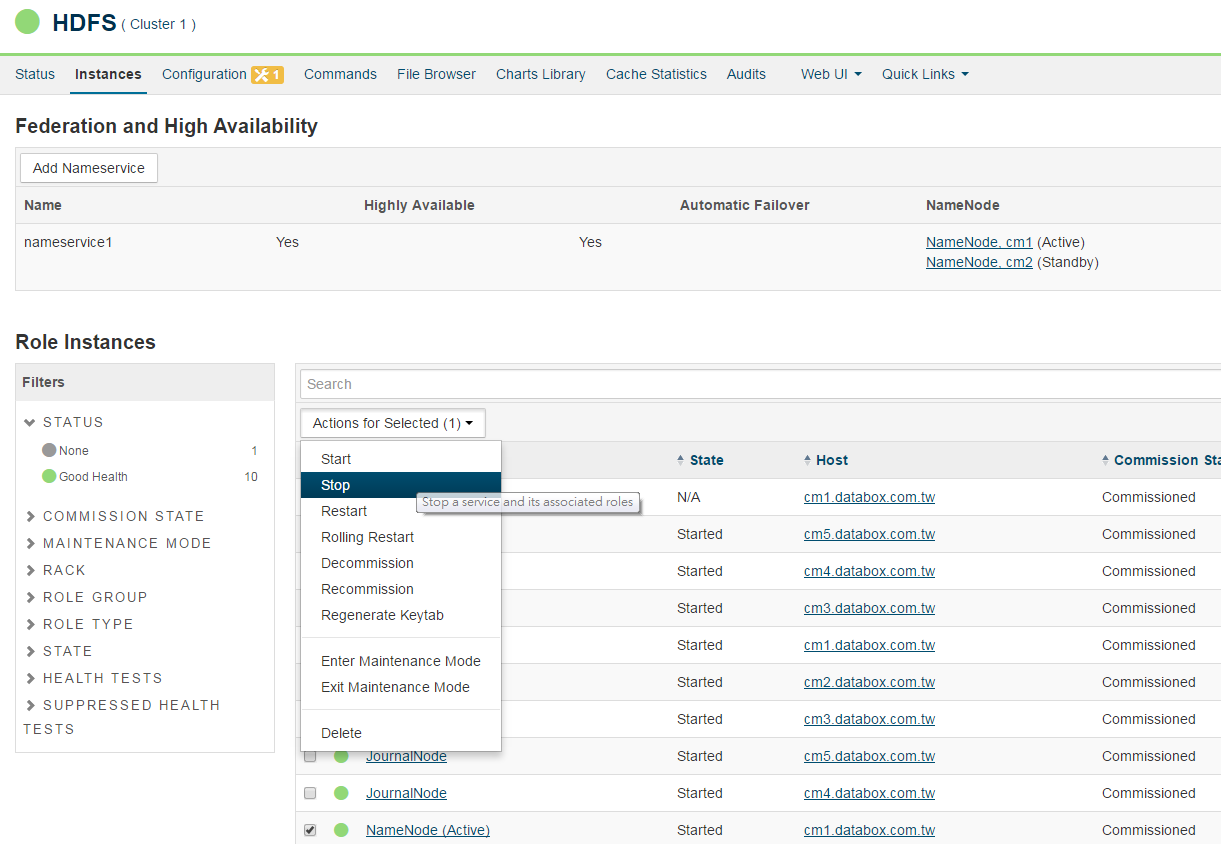
1. HDFS HA驗證

* 確認原本的狀態是為HA狀態，執行下列指令應可看到一個為active，另一個為standby

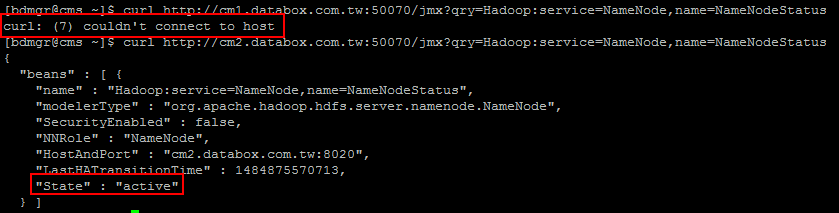
|  |
| --- |
| #> curl http://cm1.databox.com.tw:50070/jmx?qry=Hadoop:service=NameNode,name=NameNodeStatus  #> curl http://cm2.databox.com.tw:50070/jmx?qry=Hadoop:service=NameNode,name=NameNodeStatus |



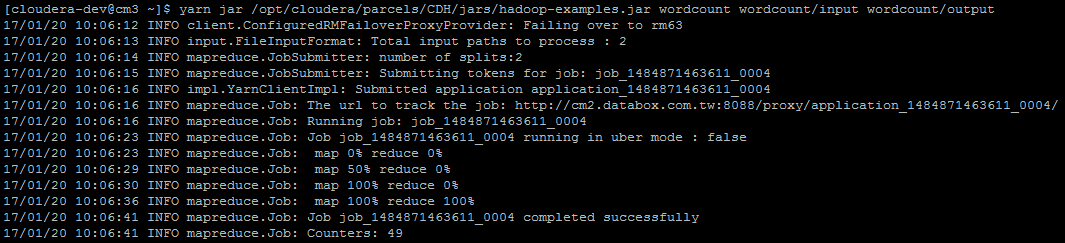
* Stop Active NameNode



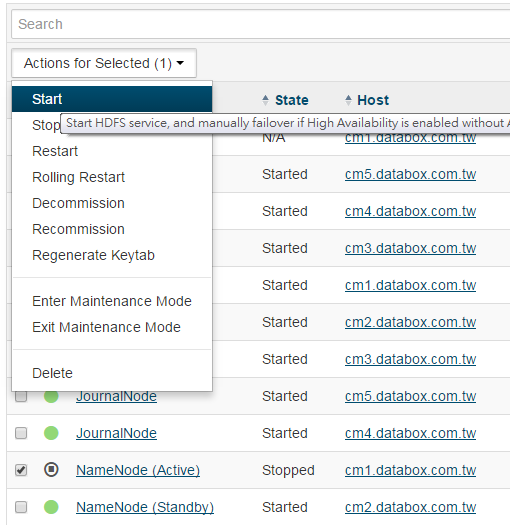
* 確認原本active的NameName(cm1)會連不到，原本standby的NameNode(cm2) state會變為active



* 執行WordCount程式是否能正常執行



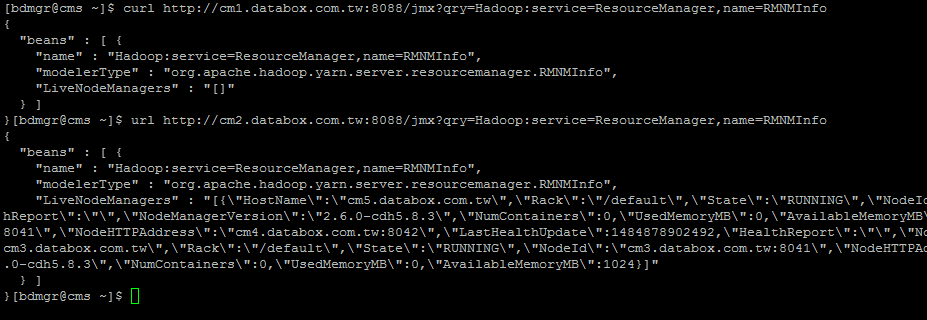
* Enable cm1 NameName



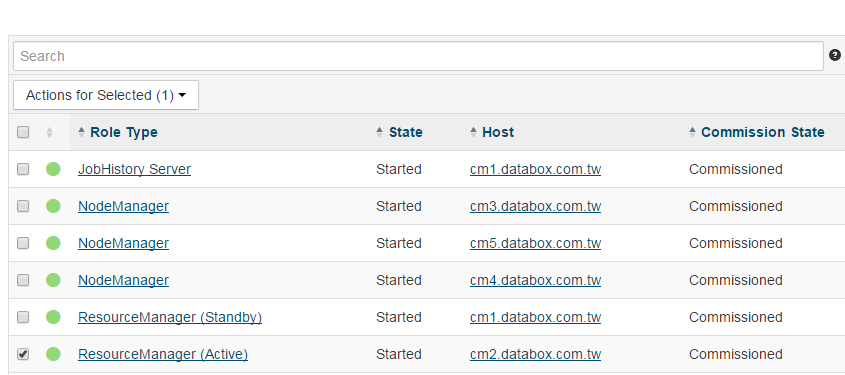
1. ResourceManager HA 驗證

* 確認原本的狀態是為HA狀態，執行下列指令應可看到active Node的LiveNodeManagers的array有值，另一個為standby Node的LiveNodeManagers的值為[]

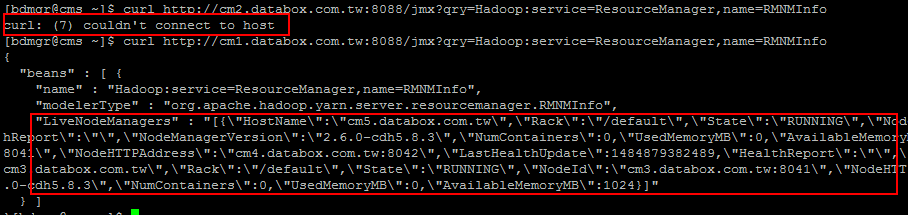
|  |
| --- |
| #> curl http://cm1.databox.com.tw:8088/jmx?qry=Hadoop:service=ResourceManager,name=RMNMInfo  #> curl http://cm2.databox.com.tw:8088/jmx?qry=Hadoop:service=ResourceManager,name=RMNMInfo |



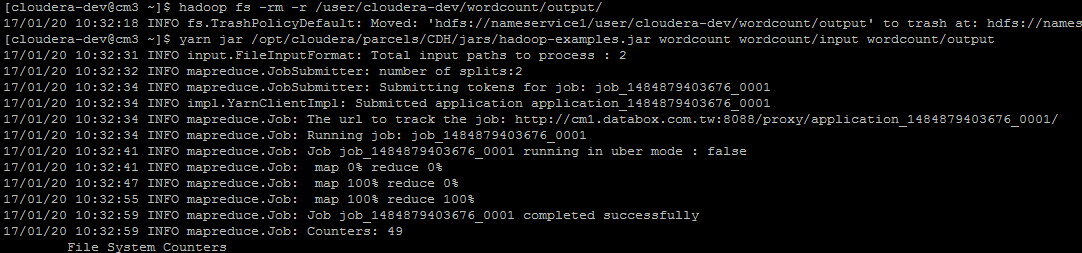
* Stop Active RM



* 確認原本active的RM(cm2)會連不到，原本standby的RM(cm1) 的LiveNodeManagers的array有值

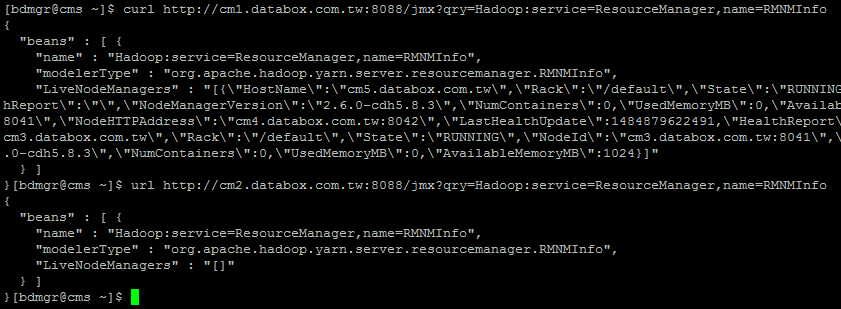


* 執行WordCount程式是否能正常執行



* Enable cm2 RM

cm2的RM重啟後會變成standby模式，cm1由原本的standby變為active



1. **Enable Kerberos**
   1. **pre chek JCE is installed**

|  |
| --- |
| #> cd $JAVA\_HOME/jre/lib/security/  #> ls local\_policy.jar  #> ls US\_export\_policy.jar  如果沒有發現上述兩檔案則須到oracle 官網下載JCE jar files，並放置至$JAVA\_HOME/jre/lib/security/目錄 |

* 1. **install kerberos server**

|  |
| --- |
| 1. install server packages by yum   #> sudo yum install krb5-server 5krb-libs krb5-auth-dialog   1. config /etc/krb5.conf, 將EXAMPLE.COM換成DATABOX.COM.TW   #> sudo vi /etc/krb5.conf  [libdefaults]  default\_realm = DATABOX.COM.TW  dns\_lookup\_realm = false  dns\_lookup\_kdc = false  ticket\_lifetime = 24h  renew\_lifetime = 7d  forwardable = true  [realms]  DATABOX.COM.TW = {  kdc = cms.databox.com.tw  admin\_server = cms.databox.com.tw  }  [domain\_realm]  .databox.com.tw = DATABOX.COM.TW  databox.com.tw = DATABOX.COM.TW   1. config /var/kerberos/krb5kdc/kdc.conf, 將EXAMPLE.COM換成DATABOX.COM.TW，並加入max\_life, max\_renewable\_life, kdc\_tcp\_ports   #> sudo vi /var/kerberos/krb5kdc/kdc.conf  [realms]  DATABOX.COM.TW = {  #master\_key\_type = aes256-cts  max\_life = 1d  max\_renewable\_life = 7d  kdc\_tcp\_ports = 88  acl\_file = /var/kerberos/krb5kdc/kadm5.acl  dict\_file = /usr/share/dict/words  admin\_keytab = /var/kerberos/krb5kdc/kadm5.keytab  supported\_enctypes = aes256-cts:normal aes128-cts:normal des3-hmac-sha1:normal arcfour-hmac:normal des-hmac-sha1:normal des-cbc-md5:normal des-cbc-crc:normal  }   1. create kerberos database   #> sudo /usr/sbin/kdb5\_util create -s   1. config the /var/kerberos/krb5kdc/kadm5.acl file   #> sudo vi /var/kerberos/krb5kdc/kadm5.acl  \*/admin@DATABOX.COM.TW \*   1. create first principal using kadmin.local   #> sudo /usr/sbin/kadmin.local -q "addprinc cloudera-scm/admin"   1. start kerberos   #> sudo service krb5kdc start  #> sudo service kadmin start  #> sudo chkconfig krb5kdc on  #> sudo chkconfig kadmin on |

* 1. **install openldap-clients on Cloudera-Manager-Sever host**

|  |
| --- |
| #> sudo yum install openldap-clients |

* 1. **install kerberos client on all hosts**

|  |
| --- |
| #> pssh -h nodes.txt -x "-t -t" -i "sudo yum install -y krb5-workstation krb5-libs krb5-auth-dialog" |

* 1. **If you have enabled YARN Resource Manager HA, you should clear the StateStore znode in ZooKeeper before enabling Kerberos**

|  |
| --- |
| 1. Go to the Cloudera Manager Admin Console home page, click to the right of the YARN service and select Stop. 2. When you see a Finished status, the service has stopped. 3. Go to the YARN service and select Actions > Format State Store.        1. When the command completes, click Close. |

* 1. **install kerberos from GUI**

|  |
| --- |
| 1. Go to the Cloudera Manager Admin Console and click to the right of the cluster for which you want to enable Kerberos authentication. 2. Select Enable Kerberos      1. double check the following steps      1. input kdc information      1. KRB5 Configuration      1. Input KDC account manager Credential   userName:cloudera-scm**/admin**     1. Import KDC account Manager Credentials Command      1. Create Kerberos Principal      1. Configure Ports      1. Enable Kerberos Commands      1. Finished |

* 1. **Create the hdfs superuser**

|  |
| --- |
| 1. change HDFS superuser group        1. Restart Stale status |

* 1. **(option) Create new user**
     + **create linux account:sasdemo**

|  |
| --- |
| #> pssh -h nodes.txt -x "-t -t" -i "sudo groupadd sasdemo"  #> pssh -h nodes.txt -x "-t -t" -i "sudo useradd -g sasdemo sasdemo" |

* + - **Add principal at KDC server**

|  |
| --- |
| #> sudo /usr/sbin/kadmin.local  #> kadmin.local: addprinc sasdemo  若只能某幾台host能用，則可以加入host  #> kadmin.local: addprinc sasdemo/host1  #> Enter password for principal "sasdemo@SAS.COM":  #> Re-enter password for principal "sasdemo@SAS.COM":  #> Principal "sasdemo@SAS.COM" created. |

* + - **Create user in HFDS /user/sasdemo**

|  |
| --- |
| #> kinit cloud-user  #> hadoop fs -mkdir /user/sasdemo  #> hadoop fs -chown sas-demo /user/sasdemo |

* + - **Run example.jar to verify sasdemo account is correct**

|  |
| --- |
| #> sudo su - sasdemo  #> kinit sasdemo  #> yarn jar /opt/cloudera/parcels/CDH/jars/hadoop-examples.jar pi 10 10000 |

* 1. **(option) Create Keytab file**

|  |
| --- |
| #> sudo /usr/sbin/kadmin.local  #> kadmin.local: xst -k sasdemo.keytab sasdemo  若只能某幾台host能用，則可以加入host  #> kadmin.local: xst -k sasdemo.keytab sasdemo/host1 sasdemo/host2  #> klist -e -k -t sasdemo.keytab  #> kdestroy  #> kinit sasdemo -k -t /etc/hadoop/conf/sasdemo.keytab  #> klist |

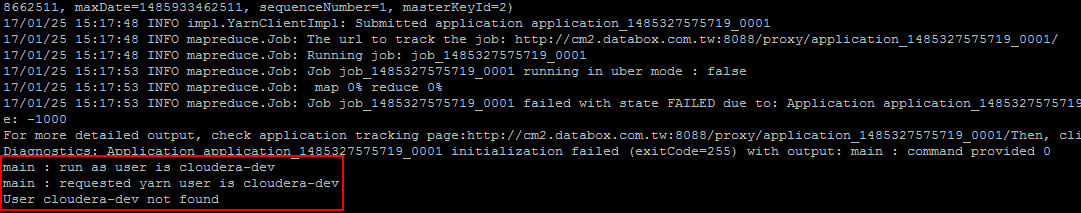
* 1. **Validation**

執行前須做下列動作

1. add cloudera-user to all nodes in Hadoop cluster

|  |
| --- |
| #> pssh -h nodes.txt -x "-t -t" -i "sudo groupadd cloudera-dev"  #> pssh -h nodes.txt -x "-t -t" -i "sudo useradd -g cloudera-dev cloudera-dev" |

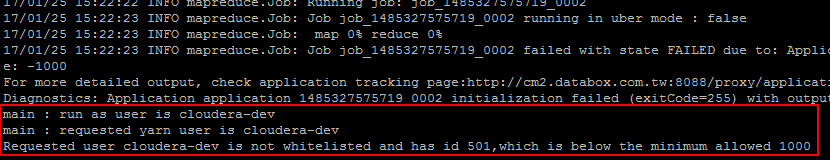
否則會出現下列錯誤



1. In yarn Configuration search : “min.user.id” and “allowed System Users”，then restart yarn

|  |
| --- |
| 1. change min.user.id to 500      1. add cloudera-dev to allowed System Users |

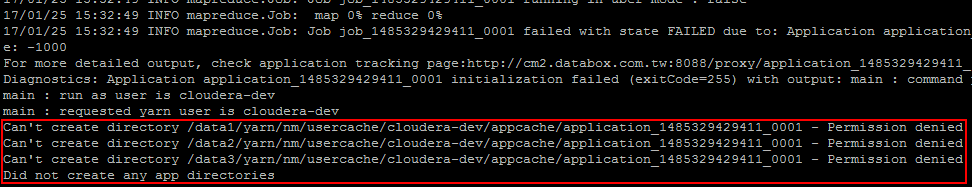
否則會出現下列錯誤



1. 執行MapReduce程式

|  |
| --- |
| #> sudo su - cloudera-dev  #> kinit cloudera-dev  #> echo “Hello Hadoop, Bye Hadoop” > file3  #> hadoop fs -put file3 /user/cloudera-dev/wordcount/input  #> hadoop fs -rmr /user/cloudera-dev/wordcount/output  #> yarn jar /opt/cloudera/parcels/CDH/jars/hadoop-examples.jar wordcount wordcount/input wordcount/output  #> hadoop fs -getmerge wordcount/output output.txt  #> cat output.txt |

若執行時發生下列錯誤，



則執行下列動作

|  |
| --- |
| #> pssh -h nodes.txt -x "-t -t" -i "sudo rm -rf /data1/yarn/nm/usercache/\*"  #> pssh -h nodes.txt -x "-t -t" -i "sudo rm -rf /data2/yarn/nm/usercache/\*"  #> pssh -h nodes.txt -x "-t -t" -i "sudo rm -rf /data3/yarn/nm/usercache/\*" |

1. **Remark**
2. **mount new disk**

若有新的hard disk加入Hadoop Cluster內，則需要做下列動作

|  |
| --- |
| 1. 找出尚未格式化的hd   #> fdisk -l   1. 以/dev/sdb為例，對/dev/sdc執行fdisk   #> sudo fdisk /dev/sdb   * 1. n   2. p   3. 1   4. Enter   5. Enter   6. w  1. 對/dev/sdc1進行格式化，並取個label name /data2   #> sudo /sbin/mkfs.ext4 -L /data2 /dev/sdc1   1. 建立掛載點   #> sudo mkdir /data2  #> sudo mount /dev/sdc1 /data2   1. 建立永久掛載點   #> sudo vi /etc/fstab  LABEL=/data2 /data2 ext4 defaults 1 2   1. 更改掛載點權限   #> sudo chown -R cloudera-scm:cloudera-scm /data2  #> sudo chmod -R 777 /data2 |

1. **bash shell commands to multi nodes**

* **use pssh**

**on line install psshl**

|  |
| --- |
| #> sudo yum install epel-release  #> sudo yum install python-pip  #> sudo pip install --upgrade pip  #> sudo pip install pssh |

**off line install psshl**

|  |
| --- |
| #> sudo yum install python-setuptools  #> wget https://storage.googleapis.ocm/google-code-archive-downloads/v2/code.google.com/parallel-ssh/pssh-2.3.1.tar.gz  #> tar zxvf pssh-2.3.1.tar.gz  #> cd pssh-2.3.1/  #> sudo python setup.py install |

**使用範例**

|  |
| --- |
| #> pssh -h nodes.txt -i "date"  ##若使用sudo 須加入 -x "-tt" 參數  #> pssh -h nodes.txt -x "-tt" -i "sudo service cloudera-scm-agent status"  ## run as root -l:ssh user -A:input password  #> pssh -h nodes.txt -l root -A -i "date"  ##copy file to remote hosts  #> pscp -h hosts.txt foo.txt /home/foo.txt  ## copy remote hosts files to local directory  #> pslurp -h hosts.txt /etc/hosts local\_dir |

* **write script by myself**

|  |
| --- |
| [bdmgr@cmgui ~]$ **cat nodes.txt**  cms  cm1  cm2  cm3  cm4  cm5  [bdmgr@cmgui ~]$ **cat mncmd.sh**  **#!/bin/bash**  **for i in $(cat nodes.txt);**  **do**  **echo "server:"$i;**  **# do your stuff here**  **ssh bdmgr@$i "hostname; date;"**  **done**  [bdmgr@cmgui ~]$ **sh mncmd.sh**  server:cms  cms.databox.com.tw  Thu Jan 19 14:23:26 CST 2017  server:cm1  cm1.databox.com.tw  Thu Jan 19 14:23:26 CST 2017  server:cm2  cm2.databox.com.tw  Thu Jan 19 14:23:26 CST 2017  server:cm3  cm3.databox.com.tw  Thu Jan 19 14:23:26 CST 2017  server:cm4  cm4.databox.com.tw  Thu Jan 19 14:23:26 CST 2017  server:cm5  cm5.databox.com.tw  Thu Jan 19 14:23:27 CST 2017  [bdmgr@cmgui ~]$ |

1. **On line MySQL Install**

此步驟使用線上安裝方式

|  |
| --- |
| ##download mysql57-community-release-el6-9.noarch.rpm to /tmp  #> sudo yum localinstall /tmp/mysql57-community-release-el6-9.noarch.rpm  #> sudo yum install mysql-server  #> sudo service mysqld start  ##從/var/log/mysqld.log找出預設的密碼  #> sudo grep 'temporary password' /var/log/mysqld.log  ##root密碼改為P@ssw0rd  #> sudo mysql\_secure\_installation  #> sudo /sbin/chkconfig mysqld on |

1. **Red Hat Enterprise 試用版註冊方式**

此步驟需access internet

|  |
| --- |
| 1. 若重新安裝另一台VM，則須在原安裝VM先刪除之前的subscribe   root #> subscription-manager remove --all   1. 在新安裝的VM，重新subscribe   root #> subscription-manager clean  root #> subscription-manager register   1. 須執行attach，才算註冊完成   root #> subscription-manager attach |

1. **failing to install Oozie ShareLib on service Oozie**

因為是使用VM，所以IO特別慢，若在first time Sartup Cluster 發生'failing to install Oozie ShareLib on service Oozie'，可將預設值得timeout時間調高

|  |
| --- |
| open another browser tab, go to console, oozie, configuration, search for oozie\_upload\_sharelib\_cmd\_timeout parameter and change it to something bigger then 270. I entered 600 |

1. **How to add new network Interface**

因為是使用VM，所以IO特別慢，若在first time Sartup Clust

|  |
| --- |
| 1. 取得network interface name   root #> nmcli   1. 在/etc/sysconfig/network-scripts/內建立一個新的ifcfg-enpXX，並且以新的interface name作為命名   root #> cp ifcfg-enps0 ifcfg-enpXXX   1. 重啟network service   root #> systemctl restart network |