securitzacio del cluster CLOUDERA

**ÍNDEX**

[1 Sigles i definicions 4](#_Toc533345979)

[1.1 Sigles 4](#_Toc533345980)

[1.2 Definicions 4](#_Toc533345981)

[2 Introducció 5](#_Toc533345982)

[3 Consideracions Inicials 6](#_Toc533345983)

[3.1 Comprovar el fitxer *config.ini* als hosts del clúster 6](#_Toc533345984)

[3.2 L'ús de Java Keystore en l'encriptació data-at-rest 6](#_Toc533345985)

[4 Encriptació de comunicacions (Data-in-transit) 7](#_Toc533345986)

[4.1 Nivell 0: Configuració bàsica TLS/SSL 7](#_Toc533345987)

[4.2 Preparar les variables d'entorn 7](#_Toc533345988)

[4.3 Obtenir i desenvoluparel Server Certificate 7](#_Toc533345989)

[4.4 Habilitar l’Encriptació TLS/SSL per Cloudera Manager Admin Console 8](#_Toc533345990)

[4.5 Nivell 1: Habilitar l’encriptació per al clúster 8](#_Toc533345991)

[4.6 Nivell 2: Habilitar els hosts de Cloudera Manager Agent per autenticar el certificat del servidor 9](#_Toc533345992)

[4.7 Nivell 3: Configurar el clúster per autenticar els certificats dels agents 10](#_Toc533345993)

[4.8 Tasques preliminars: Obtenir els certificats pels agents 10](#_Toc533345994)

[4.9 Configurar els Cloudera Manager Agents per a l'autenticació dels certificats 10](#_Toc533345995)

[4.10 Troubleshooting TLS/SSL Issues in Cloudera Manager 12](#_Toc533345996)

[5 Encriptació d'emmagatzematge (Data-at-rest) 13](#_Toc533345997)

[5.1 Preparar el clúster per a l'encriptació data-at-rest 13](#_Toc533345998)

[5.1.1 Requeriments d'entropia 13](#_Toc533345999)

[5.1.2 Instal·lar Cloudera Navigator Key Trustee Server 13](#_Toc533346000)

[5.1.3 Instal·lar Cloudera Navigator Key Trustee KMS 13](#_Toc533346001)

[5.2 HDFS Transparent Encryption 14](#_Toc533346002)

[5.2.1 Optimitzar el rendiment per a la HDFS Transparend Encryption 14](#_Toc533346003)

[4.1.1 Habilitar HDFS Encryption utilitzant el Wizard 14](#_Toc533346004)

[4.1.1.1 Afegir el servei Key Trustee Server 15](#_Toc533346005)

[4.1.1.2 Afegir el servei Key Trustee KMS 15](#_Toc533346006)

[4.1.1.3 Reiniciar els serveis obsolets i redesplegar les configuracions de client 16](#_Toc533346007)

[4.1.1.4 Validar l’encriptació de dades 16](#_Toc533346008)

[4.2 Configurar els serveis de CDH per a l’encriptació HDFS 17](#_Toc533346009)

[4.2.1 Hbase 17](#_Toc533346010)

[4.2.2 Hue 17](#_Toc533346011)

[4.2.3 MapReduce i YARN 18](#_Toc533346012)

[4.2.4 Spark 19](#_Toc533346013)

[4.2.5 Sqoop 19](#_Toc533346014)

[4.2.6 Encriptació per a aplicacions que no són serveis de CDH 20](#_Toc533346015)

[4.2.6.1 Talend 20](#_Toc533346016)

[4.2.7 Instruccions adicionals 20](#_Toc533346017)

[4.2.7.1 Configurar YARN 20](#_Toc533346018)

[5 Autenticació 21](#_Toc533346019)

[5.1 Configurar l'autenticació a Cloudera Manager i Navigator amb SAML 21](#_Toc533346020)

[5.1.1 Preparar fitxers 21](#_Toc533346021)

[5.1.2 Configurar Cloudera Manager/Navigator Metadata Server 21](#_Toc533346022)

[5.1.3 Configurar l'IDP 21](#_Toc533346023)

[5.2 Habilitar Kerberos per Cloudera 22](#_Toc533346024)

[5.2.1 Abans de començar a utilitzar el wizard 22](#_Toc533346025)

[5.2.2 KDC Information 22](#_Toc533346026)

[5.2.3 KRB5 Configuration 22](#_Toc533346027)

[5.2.4 Importar les credencials del compte de manager de KDC 22](#_Toc533346028)

[5.2.5 Configurar els ports de HDFS DataNode 23](#_Toc533346029)

[5.2.6 Enabling Kerberos 23](#_Toc533346030)

[5.2.7 Congratulations 23](#_Toc533346031)

[5.2.8 Create the HDFS Superuser 23](#_Toc533346032)

[5.2.8.1 Designating a Non-Default Superuser Group 23](#_Toc533346033)

[5.2.8.2 Executar l'script per creació d'usuaris per crear els usuaris necessaris i preparar el clúster 23](#_Toc533346034)

[5.2.8.3 Verificar que Kerberos Security funciona correctament 23](#_Toc533346035)

[5.3 Habilitar Kerberos a Kafka 24](#_Toc533346036)

[5.4 Creació de keytabs per usuaris 24](#_Toc533346037)

[5.5 Creació i assignació d'usuaris a grups 24](#_Toc533346038)

[5.6 Canvi de password d’un usuari de KDC 24](#_Toc533346039)

[5.7 HBase Authorization 25](#_Toc533346040)

[5.7.1 Entenent els nivells d'accès HBase 25](#_Toc533346041)

[5.7.2 Alguns exemples d'autorització 25](#_Toc533346042)

[5.7.3 Impersonació HBase 26](#_Toc533346043)

[5.7.4 Assignació de permisos d’accés a dades 26](#_Toc533346044)

[6 Auditoria 27](#_Toc533346045)

[6.1 Configurant Navigator Audit Server 27](#_Toc533346046)

[7 AnNexos 28](#_Toc533346047)

[7.1 Annex 1 (ajuda script create\_user.sh) 28](#_Toc533346048)

[7.1.1 Utilització amb paràmetres 28](#_Toc533346049)

[7.1.2 Utilització amb fitxer .CSV 28](#_Toc533346050)

[7.2 Annex 2 (ajuda script hbase\_auth.sh) 29](#_Toc533346051)

# Sigles i definicions

## Sigles

A continuació, trobareu una llista ordenada alfabèticament de les sigles que s'utilitzen en aquest document.

| **Terme** | **Definició** |
| --- | --- |
| CDH | Cloudera Data Hub |
| KMS | Key Management Service |
|  |  |
|  |  |
|  |  |

## Definicions

A continuació, trobareu una llista ordenada alfabèticament dels termes que s'utilitzen en aquest document.

| **Terme** | **Definició** |
| --- | --- |
| Data-at-rest | Terme que fa referència a l'encriptació de les dades del clúster. |
| Data-in-transit | Terme que fa referència a l'encriptació de les comunicacions al clúster i cap a l'exterior. |

# Introducció

Aquest document descriu i especifica les accions preses per tal de securitzar el clúster de Cloudera per una plataforma i els passos que s'han de realitzar per activar la seguretat al clúster.

La documentació està generada per a la versió 5.13.0 de CDH.

# Consideracions Inicials

Aquest és un recull de consideracions recomanades que s'haurien de realitzar abans de treballar amb el clúster per tal de facilitar els passos posteriors.

## Comprovar el fitxer *config.ini* als hosts del clúster

Comprovar que el fitxer “*config.ini”* conté la propietat **server\_host** i té assignat el nom de màquina del servidor principal de Cloudera.

## L'ús de Java Keystore en l'encriptació data-at-rest

S'ha de tenir cura de no confondre el terme “*keystore”* que fa referència a on s'emmagatzemen els certificats per l'encriptació data-in-trànsit amb el servei “*Java Keystore KMS Service”* que s'utilitza per administrar les claus per l'encriptació data-at-rest.

La relació errònia entre aquests dos artefactes del sistema pot crear la falsa sensació de la necessitat de certificats per encarar l'encriptació data-at-rest, però no és així.

La documentació de Cloudera fa referència a aquest server per donar a l'administrador la possibilitat, si s’escau, de fer una migració d'ús de la “*Java Keystore KSM Service”* cap a “*Cloudera Navigator Key Trustee Server”*. En el nostre cas, aquest pas no es requereix ja que comencem amb una configuració que no fa ús de “*Java Keystore KMS Service”*.

# Encriptació de comunicacions (Data-in-transit)

## Nivell 0: Configuració bàsica TLS/SSL

Veure: <https://www.cloudera.com/documentation/enterprise/latest/topics/cm_sg_tls_browser.html>

## Preparar les variables d'entorn

Afegir les següents línies al final dels fitxers “/etc/bashrc i ~/.bashrc”:

# export JAVA\_HOME=/usr/java/jdk1.7.0\_67-cloudera

# export JRE\_HOME=/usr/java/jdk1.7.0\_67-cloudera/jre

# export PATH=$PATH:${JAVA\_HOME}/bin:${JRE\_HOME}/bin

## Obtenir i desenvoluparel Server Certificate

Veure: <https://www.cloudera.com/documentation/enterprise/latest/topics/cm_sg_create_deploy_certs.html>

1. Crear un directori pels artefactes de seguretat
2. Crear la Java Truststore:
3. Generar les claus i els CSR
   * Servidor:

* Agents:
* Verificar la generació dels fitxers CSR:

1. Enviar el CSR a la CA
2. Verificar el certificat
3. Copiar tots els certificats retornats dins del directori /opt/cloudera/security/pki/ca per cada servidor i agents (també els certificats CA i Root)

* Convertir a PEM (només si els certificats arriben en format CER)
* Afegir Root CA i Intermediate CA a la truststore per TLS/SSL
* Convertir a PEM (només si els certificats arriben en format CER)
* Importar el certificat de root CA dins de les JDK truststores
* Importar el root CA i els certificats de CA dins de la keystore del servidor de Cloudera
* Copiar el fitxer jssecacerts del host de Cloudera Manager Server a la resta dels hosts del clúster (NOTA: s'ha de fer després d'aplicar el següent workaround):

$JAVA\_HOME/jre/lib/security/jssecacerts

* + Al host de Cloudera Manager Server, afegir el certificat intermedi de la CA al certificat signat del servidor:
    - Veure: <https://www.cloudera.com/documentation/enterprise/latest/topics/sg_add_root_ca_explicit_trust.html#id_for_this_topic>
* Importar el certificat dins la keystore:
  + Servidor:
* Importar el certificat signat de la CA i actualitzar la clau privada dins la Keystore:
* Agents:
* Importar el certificat signat de la CA i actualitzar la clau privada dins la Keystore:
* Workaround pel problema “unkown certificate”: importar el certificat del servidor dins la JDK truststore (i si és necessari, copiar-ho a tots els nodes també)

## Habilitar l’Encriptació TLS/SSL per Cloudera Manager Admin Console

Veure: <https://www.cloudera.com/documentation/enterprise/latest/topics/sg_cms_ssl.html>

1. Habilitar HTTPS per Cloudera Manager Admin Console
2. Especificar la TLS/SSL Truststore a les propietats per Cloudera Management Services
3. Reiniciar Cloudera Manager i els Serveis

## Nivell 1: Habilitar l’encriptació per al clúster

Veure: <https://www.cloudera.com/documentation/enterprise/latest/topics/cm_sg_config_tls_encr.html>

1. Habilitar l’encriptació TLS per als Cloudera Manager Agents
2. Modificar el fitxer de configuració dels agents
3. Reiniciar el Cloudera Manager Server
4. Reiniciar els Cloudera Manager Agents
5. Comprovar les comunicacions Cloudera Manager Server-Agent
6. Integrar Kerberos

## Nivell 2: Habilitar els hosts de Cloudera Manager Agent per autenticar el certificat del servidor

Veure: <https://www.cloudera.com/documentation/enterprise/latest/topics/cm_sg_config_tls_auth.html>

1. Generar un fitxer de certificat PEM encadenat, que inclourà:
   * IMI CA Certificate
   * IMI CA Root Certificate
   * Server certificate
2. Modificar el fitxer de configuració dels Cloudera Manager Agent:

* Utilitzar un editor de text per obrir el fitxer de configuració
* A la secció [Security], cercar el text [Security] i adaptar-lo
* Aplicar aquest canvi a cada fitxer de configuració dels hosts Cloudera Manager Agent

1. Reiniciar Cloudera Manager Server

* Executa la següent comanda a la consola del servidor per fer el reinici

$ sudo service cloudera-scm-server restart

1. Reiniciar els Cloudera Manager Agents

* Per cada agent, executa la següent comanda:

$ sudo service cloudera-scm-agent restart

1. Comprovar les comunicacions Cloudera Manager Server-Agent:
   1. Obrir la Cloudera Manager Admin Console.
   2. Seleccionar Hosts > All Hosts.
   3. Obrir el filtre Last Heartbeat per veure el seu estat. Aquest estat hauria de ser Good, significant que el servidor i els agents es comuniquen de forma correcta.

## Nivell 3: Configurar el clúster per autenticar els certificats dels agents

Veure la documentació de Cloudera: <https://www.cloudera.com/documentation/enterprise/latest/topics/cm_sg_config_tls_agent_auth.html>

## Tasques preliminars: Obtenir els certificats pels agents

Aquest pas ha sigut completat al nivell 0.

1. Crear els directoris pels artefactes de seguretat
2. Generar les claus i els CSRs per als Cloudera Manager Agent Hosts
3. Enviar els CSRs a la CA
4. Verificar els certificats

La resposta hauria d'incloure TLS Web Server Authentication i TLS Web Client Authentication, com es mostra sota:

X509v3 Extended Key Usage:

TLS Web Server Authentication, TLS Web Client Authentication

1. Distribuir els certificats als Cloudera Manager Agents:

* Recordar, utilitzar la keystore generada al Nivell 0
* Crear el certificat encadenat

1. Crear un link simbòlic al certificat i la keystore

## Configurar els Cloudera Manager Agents per a l'autenticació dels certificats

1. Importar els certificats CA i CA Root a la keystore
2. Importar els certificats de resposta de la CA i actualitzar la clau privada dins la keystore
3. Exportar la clau privada a un fitxer:
   1. Creat una versió PKCS12 de la Java kestore:

* Per a l'agent del servidor
* Per a la resta d'agents

1. Utilitzar la comanda openssl per exportar la clau privada al seu propi fitxer
2. Crear un link simbòlic per al fitxer .key
3. Crear un fitxer password
4. Configurar els agents per utilitzar les claus privades i els certificats:

* Utilitzar un editor de text per obrir el fitxer de configuració al següent path:
* A la secció [Security], cercar el detall dels certificats PEM:
  + - client\_key\_file, que identifica el path per a la clau privada.
    - client\_keypw\_file, que identifica el path per al fitxer del password de la clau privada.
    - client\_cert\_file, que identifica el path per al certificat del client.
  + Eliminar la marca de hash(#) de les propietats i deixar la configuració com es mostra a continuació:

[security]

...

# PEM file containing client private key

client\_key\_file=/opt/cloudera/security/pki/agent.key

...

# If client\_keypw\_cmd isn't specified, instead a text file contining

# the client private key password can be used.

client\_keypw\_file=/etc/cloudera-scm-agent/agentkey.pw

# PEM file containing client certificate.

client\_cert\_file=/opt/cloudera/security/pki/ca/agent.cert.pem ...

1. Habilitar l'autenticació dels certificats d'agent:
   1. Entra a la Cloudera Manager Admin Console.
   2. Seleccionar Administration > Settings.
   3. Clicar la categoria Security.
   4. Configurar les següents propietats TLS:

* Use TLS Authentication of Agents to Server: marca la casella.
* Cloudera Manager TLS/SSL Certificate Trust Store File : $JAVA\_HOME/jre/lib/security/jssecacerts
* Cloudera Manager TLS/SSL Certificate Trust Store Password: changeit
  1. Clicar Save Changes.

1. Reiniciar Cloudera Manager Server i Agents

* Al servidor:

# sudo service cloudera-scm-server restart

* A cada agent:

# sudo service cloudera-scm-agent restart

1. Verify Cloudera Manager Server and Agent Communications

## Troubleshooting TLS/SSL Issues in Cloudera Manager

Veure: <https://www.cloudera.com/documentation/enterprise/latest/topics/cm_sg_tls_troubleshoot.html#import_cloudera_support_cert_pubkey>

# Encriptació d'emmagatzematge (Data-at-rest)

## Preparar el clúster per a l'encriptació data-at-rest

### Requeriments d'entropia

Per comprovar l'entropia s'ha d'executar la següent comanda:

# cat /proc/sys/kernel/random/entropy\_avail

Si el valor està per sota de 500 s'ha d'augmentar instal·lant *rng\_tools* de la següent manera:

Provem si tenim RHEL6 o RHEL7

$ sudo cat /etc/\*-release

Per CentOS 6:

$ sudo yum install rng-tools

$ sudo echo 'EXTRAOPTIONS="-r /dev/urandom"' >> /etc/sysconfig/rngd

$ sudo service rngd start

$ sudo chkconfig rngd on

Per CentOS 7:

$ sudo yum install rng-tools

$ cp /usr/lib/systemd/system/rngd.service /etc/systemd/system/

$ sed -i -e 's/ExecStart=\/sbin\/rngd -f/ExecStart=\/sbin\/rngd -f -r \/dev\/urandom/' /etc/systemd/system/rngd.service

$ systemctl daemon-reload

$ systemctl start rngd

$ systemctl enable rngd

### Instal·lar Cloudera Navigator Key Trustee Server

Per activar la parcel de *Cloudera Navigator Key Trustee Server* s'han de seguir les següents passes:

1. Seleccionar la versió concreta com a *Parcel* a la pàgina de descarrega (<https://www.cloudera.com/downloads/navigator/key-trustee-server/5-13-0.html>)
2. Extreure i copiar *KEYTRUSTEE\_SERVER-5.13.0-1.keytrustee5.13.0.p0.6-el7.parcel i KEYTRUSTEE\_SERVER-5.13.0-1.keytrustee5.13.0.p0.6-el7.parcel.sha1* al directori */opt/cloudera/parcel-repo* i accedeix a ell des de la línia de comandes
3. Crear un hash SHA1 per a la parcel afegida i guardar-lo en un fitxer anomenat *parcel\_name.parcel.sha*

# sha1sum KEYTRUSTEE\_SERVER-5.13.0-1.keytrustee5.13.0.p0.6-el7.parcel | awk '{ print $1 }' > KEYTRUSTEE\_SERVER-5.13.0-1.keytrustee5.13.0.p0.6-el7.parcel.sha

1. A Cloudera Manager navegar a la pàgina de Parcels
2. Polsar sobre ***Check for New Parcels***
3. ***Download***, ***Distribute*** i ***Activate*** el parcel *KEYTRUSTEE\_SERVER*

### Instal·lar Cloudera Navigator Key Trustee KMS

Per activar la parcel *Cloudera Navigator Key Trustee KMS* s'han de seguir les següents passes:

1. Seleccionar la versió concreta com a parcel a la pàgina de descarrega: (<https://www.cloudera.com/downloads/navigator/key-trustee-kms/5-13-0.html>)
2. Extreure i copiar *KEYTRUSTEE-5.13.0-1.KEYTRUSTEE5.13.0.p0.2-el7.parcel i KEYTRUSTEE-5.13.0-1.KEYTRUSTEE5.13.0.p0.2-el7.parcel.sha1* al directori */opt/cloudera/parcel-repo* i accedir-hi a ell des de la línia de comandes.
3. Crear un hash SHA1 per la parcel afegida i guardar-lo en un fitxer anomenat *parcel\_name.parcel.sha*

# sha1sum KEYTRUSTEE-5.13.0-1.KEYTRUSTEE5.13.0.p0.2-el7.parcel | awk '{ print $1 }' > KEYTRUSTEE-5.13.0-1.KEYTRUSTEE5.13.0.p0.2-el7.parcel.sha

1. A Cloudera Manager navegar a la pàgina de Parcels
2. Polsar sobre ***Check for New Parcels***
3. ***Download***, ***Distribute*** i ***Activate*** el parcel *KEYTRUSTEE*

## HDFS Transparent Encryption

### Optimitzar el rendiment per a la HDFS Transparend Encryption

1. Instal·lar el paquet openssl-devel a tots els hosts del clúster:

$ sudo yum install openssl-devel

1. Per verificar que un host està llest per utilitzar el set d'instruccions d'optimització AES-NI per HDFS encryption at rest, utilitzar la següent comanda:

$ hadoop checknative

1. S'ha de rebre la següent sortida:

[cloudera@quickstart ~]$ hadoop checknative

17/11/30 00:32:14 INFO bzip2.Bzip2Factory: Successfully loaded & initialized native-bzip2 library system-native

17/11/30 00:32:14 INFO zlib.ZlibFactory: Successfully loaded & initialized native-zlib library

Native library checking:

hadoop: true /opt/cloudera/parcels/CDH-5.12.0-1.cdh5.12.0.p0.29/lib/hadoop/lib/native/libhadoop.so.1.0.0

zlib: true /lib64/libz.so.1

snappy: true /opt/cloudera/parcels/CDH-5.12.0-1.cdh5.12.0.p0.29/lib/hadoop/lib/native/libsnappy.so.1

lz4: true revision:10301

bzip2: true /lib64/libbz2.so.1

openssl: true /usr/lib64/libcrypto.so

1. Si la línia de openssl està a **true**, llavors Hadoop ha detectat la versió correcta de **libcrypto.so** i la optimització funciona.

### Habilitar HDFS Encryption utilitzant el Wizard

Accedir a ***Administration > Security > Set up HDFS Data At Rest Encryption***

A la primera pàgina del wizard, seleccionar l'arrel de confiança per les claus d'encriptació:

* **Cloudera Navigator Key Trustee Server**

A continuació, s'han de validar les següents passes:

1. Habilitar Kerberos
2. Habilitar TLS/SSL
3. Instal·lar Key Trustee Server
4. Instal·lar Key Trustee KMS
5. Afegir el servei Key Trustee Server
6. Afegir el servei Key Trustee KMS
7. Reiniciar els serveis obsolets i tornar a desplegar les configuracions de client
8. Validar l'encriptació de dades

#### Afegir el servei Key Trustee Server

Per afegir el nou servei s'han de seguir les següents passes del wizard:

1. Acceptar les responsabilitats
2. Seleccionar un host
3. Seleccionar el primer candidat
4. Seleccionar el candidat passiu
5. Selecció final
6. Comprovar l'entropia
7. Sincronitzar claus dels Key Trustee Server Actiu i Passiu
8. No fer res perquè ja tenim el certificats signats
9. Continuar amb les següents configuracions per defecte
10. Iniciar el servei Key Trustee Server
11. Finalitzar

#### Afegir el servei Key Trustee KMS

Per afegir el nou servei s'han de seguir les següents passes del wizard:

1. Seleccionar Existing Key Trustee Server i omplir els camps amb els noms dels hosts del servidor actiu i pasiu
2. Seleccionar rcosclmr10c.imi.bcn amb el nostre host del KMS
3. Comprovar la entropia al nostre host de KMS
4. Crear l’organització (Ajuntament)
5. Configurar usuaris y Llistes de Control d'Access (ACL)
6. Continuar amb els valors per defecte al apartat Review Changes
7. Iniciar el servei Key Trustee KMS
8. Continuar fins a finalitzar

#### Reiniciar els serveis obsolets i redesplegar les configuracions de client

Aplicar els canvis i reiniciar el serveis obsolets.

#### Validar l’encriptació de dades

Executar les següents línies de comandes:

[root@rcosclmr08a ~]# kinit cloudera-scm/admin

Password for cloudera-scm/admin@CLOUDERA:

[root@rcosclmr08a ~]# hadoop key create mykey1

mykey1 has been successfully created with options Options{cipher='AES/CTR/NoPadding', bitLength=128, description='null', attributes=null}.

[root@rcosclmr08a ~]# hadoop fs -mkdir /tmp/zone1

[root@rcosclmr08a ~]# kadmin

Authenticating as principal cloudera-scm/admin@CLOUDERA with password.

Password for cloudera-scm/admin@CLOUDERA:

kadmin: addprinc hdfs@CLOUDERA

WARNING: no policy specified for hdfs@CLOUDERA; defaulting to no policy

Enter password for principal "hdfs@CLOUDERA":

Re-enter password for principal "hdfs@CLOUDERA":

Principal "hdfs@CLOUDERA" created.

kadmin: exit

[root@rcosclmr08a ~]# kinit hdfs

[root@rcosclmr08a ~]# hdfs crypto -createZone -keyName mykey1 -path /tmp/zone1

Added encryption zone /tmp/zone1

[root@rcosclmr08a ~]# kinit cloudera-scm/admin@CLOUDERA

Password for cloudera-scm/admin@CLOUDERA:

[root@rcosclmr08a ~]# echo "Hello World" > /tmp/helloWorld.txt

[root@rcosclmr08a ~]# hadoop fs -put /tmp/helloWorld.txt /tmp/zone1

[root@rcosclmr08a ~]# hadoop fs -cat /tmp/zone1/helloWorld.txt

Hello World

[root@rcosclmr08a ~]# rm /tmp/helloWorld.txt

[root@rcosclmr08a ~]# kinit hdfs

Password for hdfs@CLOUDERA:

[root@rcosclmr08a ~]# hadoop fs -cat /.reserved/raw/tmp/zone1/helloWorld.txt

v,+▒▒ ▒▒▒[root@rcosclmr08a ~]#

[root@rcosclmr08a ~]# hadoop fs -rm -R /tmp/zone1

17/12/05 16:36:47 INFO fs.TrashPolicyDefault: Moved: 'hdfs://nameservice1/tmp/zone1' to trash at: hdfs://nameservice1/user/hdfs/.Trash/Current/tmp/zone1

## Configurar els serveis de CDH per a l’encriptació HDFS

### Hbase

A les llistes d'accés de KMS donar a l'usuari hbase i el grup hbase el permís DECRYPT\_EEK per a la clau d'Hbase:

<property>

<name>key.acl.hbase-key.DECRYPT\_EEK</name>

<value>hbase hbase</value>

</description>

</property>

També es poden editar a **Key Trustee KMS > Configuration > Key Management Server Proxy Advanced Configuration Snippet (Safety Valve) for kms-acls.xml**.

* Detenir el servei Hbase. (Aquesta aturada també pot parar Hue i Spark)
* Moure dades des del directori /hbase cap a /hbase-tmp *(Copia de seguretat)*
* Crear un directori buit /hbase i crear la zona d'encriptació:
* Distcp totes les dades de /hbase-tmp cap a /hbase, preservant els permisos usuari-grup i atributs estesos
* Iniciar el servei Hbase i verificar que està funcionant segons l'esperat.
* Eliminar el directori /hbase-tmp

### Hue

A les llistes d'accés de KMS, donar tant als usuaris oozie i hue com als grups oozie i hue, el permís DECRYPT\_EEK per la clau d'HUE:

<property>

<name>key.acl.hbase-key.DECRYPT\_EEK</name>

<value>hbase hbase</value>

</description>

</property>

També es poden editar a **Key Trustee KMS > Configuration > Key Management Server Proxy Advanced Configuration Snippet (Safety Valve) for kms-acls.xml**.

* Crear un directori buit /user/hue-tmp

[root@rcosclmr10c ~]# hdfs dfs -mkdir /user/hue-tmp

* Fer /user/hue-tmp una zona d'encriptació

[root@rcosclmr10c ~]# kinit cloudera-scm/admin

[root@rcosclmr10c ~]# hadoop key create hue-key

[root@rcosclmr10c ~]# kinit hdfs

[root@rcosclmr10c ~]# hdfs crypto -createZone -keyName hue-key -path /user/hue-tmp

* Distcp totes les dades de /user/hue cap a /user/hue-tmp

[root@rcosclmr10c ~]# kinit hue

[root@rcosclmr10c ~]# hdfs dfs -chmod a+rwx /user/hue-tmp

[root@rcosclmr10c ~]# hdfs dfs -chown hue:hue /user/hue-tmp

[root@rcosclmr10c ~]# kinit hdfs

[root@rcosclmr10c ~]# hdfs dfs -chown hue:hue /user/hue-tmp/.Trash

[root@rcosclmr10c ~]# kinit hue

[root@rcosclmr10c ~]# hadoop distcp -p -skipcrccheck -update hdfs://nameservice1:8020/user/hue/ hdfs://nameservice1:8020/user/hue-tmp

* Eliminar /user/hue i canviar el nom a /user/hue-tmp a /user/hue

[root@rcosclmr10c ~]# hdfs dfs -mv /user/hue /user/hue-bak

[root@rcosclmr10c ~]# hdfs dfs -mv /user/hue-tmp /user/hue

[root@rcosclmr10c ~]# hadoop fs -rm -R /user/hue-bak

### MapReduce i YARN

Fer /user/history una única zona d'encriptació, perquè els fitxers d'història són moguts entre els directoris *intermediate* i *done*, i l’encriptació HDFS no permet moure fitxers encriptats a través de zones d'encriptació. Quan es crea la zona d'encriptació nomenar la clau **mapred-key** per aprofitar les ACLs autogenerades.

Les ACLs de KMS han de permetre els usuaris mapred i yarn; i el grup hadoop pel permís DECRYPT\_EEK. Aquest valor es pot editar a **Key Trustee KMS > Configuration > Key Management Server Proxy Advanced Configuration Snippet (Safety Valve) for kms-acls.xml** i el valor que ha d'agafar és el següent:

Name: key.acl.mapred-key.DECRYPT\_EEK

Value: mapred,yarn hadoop

* Si /user/history no existeix, crear el directori i fer-ho una zona d'encriptació.
* Si /user/history existeix, moure /user/history a /user/history-bak.
* Crear un nou directori /user/history i fer-ho una zona d'encriptació:

[root@rcosclmr10c ~]# kinit cloudera-scm/admin

[root@rcosclmr10c ~]# hadoop key create mapred-key

[root@rcosclmr10c ~]# kinit mapred

[root@rcosclmr10c ~]# hdfs dfs -mv /user/history /user/history-bak

[root@rcosclmr10c ~]# hdfs dfs -mkdir /user/history

[root@rcosclmr10c ~]# kinit hdfs

[root@rcosclmr10c ~]# hdfs crypto -createZone -keyName mapred-key -path /user/history

* Distcp totes les dades de /user/history-bak cap a /user/history i crear l'estructura de fitxers necessària:

[root@rcosclmr10c ~]# hdfs dfs -mkdir /user/history/done

[root@rcosclmr10c ~]# hdfs dfs -chmod 770 /user/history/done

[root@rcosclmr10c ~]# hdfs dfs -mkdir /user/history/done\_intermediate

[root@rcosclmr10c ~]# hdfs dfs -chmod 777 /user/history/done\_intermediate

* Eliminar /user/history-bak després de comprovar que l’encriptació funciona com l'esperat.

### Spark

Actualitzar les ACLs del KMS a **Key Trustee KMS > Configuration > Key Management Server Proxy Advanced Configuration Snippet (Safety Valve) for kms-acls.xml** amb el següent valor:

Name: key.acl.spark-key.DECRYPT\_EEK

Value: spark spark

Executar les següents comandes al host del KMS

[root@rcosclmr10c ~]# kinit cloudera-scm/admin

[root@rcosclmr10c ~]# hadoop key create spark-key

[root@rcosclmr10c ~]# kinit hdfs

[root@rcosclmr10c ~]# hdfs crypto -createZone -keyName spark-key -path /user/spark/applicationHistory

### Sqoop

Actualitzar les ACLs del KMS a **Key Trustee KMS > Configuration > Key Management Server Proxy Advanced Configuration Snippet (Safety Valve) for kms-acls.xml** amb el següent valor:

Name: key.acl.sqoop-key.DECRYPT\_EEK

Value: sqoop sqoop

### Encriptació per a aplicacions que no són serveis de CDH

#### Talend

Actualitzar les ACLs del KMS a **Key Trustee KMS > Configuration > Key Management Server Proxy Advanced Configuration Snippet (Safety Valve) for kms-acls.xml** amb el següen valor:

Name: key.acl.talend-key.DECRYPT\_EEK

Value: talend talend

### Instruccions adicionals

#### Configurar YARN

* Redir el user id mínim:

1. Anar a Yarn > Configuration
2. Cercar: \_ Minimum User ID\_
3. Posar el valor 900
4. Guardar els canvis
5. Reiniciar el servei Yarn

* Opció *Always Use Linux Container Executor*

1. Anar a Yarn > Configuration
2. Cercar: Always Use Linux Container Executor
3. Activar el checkbox
4. Guardar canvis

* Afegir Yarn Gateways

1. Per executar la comanda distcp en instàncies hdfs, es necessita afegir dos yarn gateways: un per cada hdfs namenode
2. Després d'això, s'ha de tornar a la pàgina principal i entrar a la instancia de YARN
3. Redesplegar les configuracions pendents del client

# Autenticació

## Configurar l'autenticació a Cloudera Manager i Navigator amb SAML

### Preparar fitxers

És necessari preparar els següents fitxers i informació i configurar Cloudera Manager i Cloudera Navigator:

* Una Java keystore que conté una clau privada per Cloudera Manager/Navigator per signar /encriptar els missatges SAML.
* El fitxer XML de metadades de SAML del IDP.
* L’entity ID que s'utilitzarà per identificar la instància de Cloudera Manager/Navigator.
* Com es passa el user ID als missatges SAML, al nostre cas s'utilitza NameID.
* El mètode que establirà el rol de Cloudera Manager/Navigator de l'usuari, al nostre cas s'utilitzarà un atribut. Necessitem el nom de l'atribut.Autorització.

### Configurar Cloudera Manager/Navigator Metadata Server

1. Entrar a la Cloudera Manager Admin Console
2. Seleccionar Administration > Settings (Cloudera Manager) o Cluster > Cloudera Management Service (Cloudera Navigator)
3. Seleccionar Configuration (Cloudera Navigator)
4. Seleccionar External Authentication com a filtre de categoria.
5. Posar la propietat External Authentication Type a SAML.
6. Posar la propietat Path to SAML IDP Metadata File a /opt/cloudera/security/saml/RCOSSECU03A\_saml2\_metadata.xml
7. Posar la propietat Path to SAML Keystore File a /opt/cloudera/security/saml/RCOSSECU03A.jks
8. Posar la propietat SAML Keystore Password a xxxxx
9. Posar la propietat Alias of SAML Sign/Encrypt Private Key a RCOSSECU03A
10. Posar la propietat SAML Sign/Encrypt Private Key Password a xxxxx
11. Posar la propietat SAML Entity ID a cloudera.manager/cloudera.navigator
12. Posar la propietat SAML Entity Alias a cloudera.manager/cloudera.navigator
13. Posar la propietat SAML Response Binding a HTTP-Post
14. Posar la propietat Source of User ID in SAML Response a NameID
15. Posar la propietat SAML Role assignment mechanism a Attribute
16. Posar la propietat SAML attribute identifier for user role a [http://wso2.org/claims/role](http://wso2.org/claims/cmrole)
17. Guarda els canvis.
18. Reïniciar el Cloudera Management Service.

### Configurar l'IDP

1. Descarregar el fitxer de metadades de SAML de Cloudera desde <http://rcosclmr08a.imi.bcn:7180/saml/metadata>.
2. S’obtenen les següents URL per a l'IDP:
   * <SingleLogoutService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST" Location="https://rcosclmr08a.imi.bcn:7183/saml/SingleLogout/alias/cloudera.manager"> for Single Logout URL
   * <AssertionConsumerService Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST" Location="https://rcosclmr08a.imi.bcn:7183/saml/SSO/alias/cloudera.manager" index="1" /> for ACS URL
3. Assegurar que l'IDP té accés a qualsevol certificat públic necessari per validar les claus privades, es poden exportar amb la següent comanda:

keytool -export -alias rcosclmr08a.imi.bcn -keystore rcosclmr08a.imi.bcn-server.jks -file rcosclmr08a\_public.pem

## Habilitar Kerberos per Cloudera

Per iniciar el wizard:

* Anar a Cloudera Manager Admin console i clicar al desplegable de la dreta del clúster al qual es vol habilitar Kerberos
* Seleccionar Enable Kerberos

### Abans de començar a utilitzar el wizard

La pàgina de benvinguda mostra les passes que s'han de completar abans de començar el wizard.

* Instal·lar openldap-clients al host Cloudera Manager Server
* Instal·lar krb5-workstation, krb5-libs a tots els hosts
* Crear un compte per Cloudera Manager que tingui permisos per crear altes comptes al KDC

kadmin: addprinc -pw xxxxx [cloudera/admin@[platform\_name].IMI.BCN](mailto:cloudera/admin@%5bplatform_name%5d.IMI.BCN)

Un cop es puguin marcar totes les capses, clicar Continue.

### KDC Information

Seleccionar MIT KDC i clicar Continue.

### KRB5 Configuration

Clicar Continue.

### Importar les credencials del compte de manager de KDC

Introduir el nom d'usuari i el password per l'usuari que pot crear principals pel clúster CDH al KDC.

Clicar Continue.

### Configurar els ports de HDFS DataNode

Marcar la casella per confirmar el reinici del clúster. Clicar Continue.

### Enabling Kerberos

Esperar fins que el procés finalitzi i clicar Continue.

### Congratulations

Clicar Finish per tornar a la pàgina principal de la Cloudera Manager Admin Console.

### Create the HDFS Superuser

#### Designating a Non-Default Superuser Group

Per designar un grup diferent de superusers en comptes del que hi ha per defecte per HDFS, seguir les següents passes:

1. Anar a la Cloudera Manager Admin Console i navegar al servei HDFS.
2. Clicar la pestanya Configuration.
3. Seleccionar Scope > HDFS (Service-Wide).
4. Seleccionar Category > Security.
5. Localitzar la propietat Superuser Group i canviar el valor al nom del grup apropiat per l’entorn desitjat, per exemple, supergroup.
6. Clicar Save Changes.
7. Reiniciar el servei HDFS.

Per habilitar l'accés al compte de superuser s'ha de crear un principal de Kerberos amb el component principal supergroup:

kadmin: addprinc supergroup

# kinit supergroup

#### Executar l'script per creació d'usuaris per crear els usuaris necessaris i preparar el clúster

Executar l'script /opt/cloudera/scripts/create\_user.sh seguint l'ajuda a l'annex 1.

#### Verificar que Kerberos Security funciona correctament

$ kinit USERNAME@YOUR-LOCAL-REALM.COM

$ hadoop jar /opt/cloudera/parcels/CDH/lib/hadoop-0.20-mapreduce/hadoop-examples.jar pi 10 10000

Number of Maps = 10

Samples per Map = 10000

...

Job Finished in 30.958 seconds

Estimated value of Pi is 3.14120000000000000000

## Habilitar Kerberos a Kafka

NOTA: Aquesta informació s'ha extret de https://www.cloudera.com/documentation/kafka/latest/topics/kafka\_security.html

Per habilitar l’a autentiació Kerberos per Kafka:

• Des de Cloudera Manager, navegar a Kafka > Configuration. Posar SSL client authentication a None. Posar Inter Broker Protocol a SASL\_PLAINTEXT.

• Clicar Save Changes.

• Reiïniciar el servei Kafka.

• Assegurar-se de que listeners = SASL\_PLAINTEXT està present als logs del Kafka Broker /var/log/kafka/kafka-broker-rcosclmr10c.imi.bcn.log

## Creació de keytabs per usuaris

Per crear keytabs per nous usuaris fa falta executar les següents pases:

• Accedir a kadmin.local desde el host que conté el KDC

• Dins de kadmin.local exeutar la comanda xst amb els paràmetres necessaris

# kadmin.local

kadmin.local: xst -k keytabfilename.keytab principal\_name

Això reiniciarà el password pel principal seleccionat i crearà un fitxer amb les credencials d'aquel principal. Si el que es vol es mantenir el password s'haurà de completar la crida amb el paràmetre **norandkey** com s'indica a continuació

#kadmin.local

kadmin.local: xst -norandkey -k keytabfilename.keytab principal\_name

## Creació i assignació d'usuaris a grups

Per tal de fer la gestió dels usuaris i crear de nous per seguir amb el procés de creació que s'inclou a la documentació de Cloudera s'ha creat un script al path **/opt/cloudera/scripts/create\_user.sh** del **Master 1** als clústers de PRE i PRO que s'encarrega de fer totes les passes de la creació dels usuaris. El manual d'us d'aquest script es pot trobar a l'[Anex 1](#_4fsjm0b) d'aquest document.

## Canvi de password d’un usuari de KDC

Per canviar el password d’un usuari al KDC, des de la consola del host que te el Kerberos instal·lat, primer s’haura d’accedir a la consola d’administració de Kerberos *kadmin.local*.

Un cop dins només s’ha d’executar la comanda *change\_password* ó *cpw.*

*kadmin.local: cpw principal*

*Enter password for principal principal@CLOUDERA: <= Type the new password.*

*Re-enter password for principal principal@CLOUDERA: <= Type it again.*

*Password for* [*principal@CLOUDERA*](mailto:principal@CLOUDERA) *changed.*

*kadmin.local:*

Autorització

## HBase Authorization

NOTA: passos extrets des de <https://www.cloudera.com/documentation/enterprise/5-12-x/topics/cdh_sg_hbase_authorization.html>

### Entenent els nivells d'accès HBase

Els nivells d'accés es concedeixen independentment els uns dels altres i permeten diferents tipus d'operació a un abast donat.

* Read (R) – Pot llegir
* Write (W) – Pot escriure
* Execute (X) – Pot executar
* Create (C) – Pot crear i eliminar taules incloent les que no son les creades per si mateix.
* Admin (A) – Pot realitzar operacions al clúster com balancejar el clúster o assignar regions.

Els abasts possibles són:

* Superuser – els superusers poden realitzar qualsevol operació disponible en Hbase, a qualsevol recurs.
* Global – els permisos concedits a un abast global permeten a l'administrador operar a totes les taules del clúster.
* Namespace – els permisos concedits a aquest abast apliquen a totes les taules d'un namespace donat.
* Table – els permisos concedits a l'abast de taula només apliquen a les dades o metadades a una taula donada.
* ColumnFamily – els permisos concedits a aquest abast aplica a cel·les dins d'una ColumnFamily.
* Cell – els permisos concedits a aquest abast apliquen a, exactament, la cel·la donada.

### Alguns exemples d'autorització

Per tal de donar permís a grups o usuaris a artefactes d'hbase primer s'haurà d'entrar al hbase-shell desde la linia de comandes. Aquí es mostraran alguns exemples d'us.

**Sintaxi de la crida**

grant <user>, <permissions> [, <@namespace> [, <table> [, <column family> [, <column qualifier>]]]

**Assignar un grup a una taula permisos de lectura i escriptura**

grant ['@g](about:blank)roup\_name', 'RW', 'table\_name'

**Assignar un grup a un namespace amb permisos de lectura i escriptura**

grant ['@group](about:blank)\_name', 'RW', ['@namespace](about:blank)'

**Assignar un usuari a una taula amb permisos només de lectura**

grant 'user\_name', 'R', 'table\_name'

### Impersonació HBase

Per habilitar la impersonació en HBase per utilitzar la aplicació d'HBase per HUE, s’ha de seguir les següents passes:

1. Anar al servei HBase a Cloudera Manager
2. Clicar la pestanya Configuration
3. A la capsa de cerca escriure thrift
4. Marcar les següents propietats:
   1. Enable Hbase Thrift Http Server
   2. Enable Hbase Thrift Proxy Users
5. Guardar els canvis
6. Reiniciar el servei

### Assignació de permisos d’accés a dades

L’assignació de dels permisos sobre les diferents taules d’HBase s’aconsegueix configurant el el fitxer de dades ***hbase\_auth.csv***.

Cada linia d’aquest fitxer té un format del tipus:

**nom\_taula;permis;grups**

On:

* ***nom\_taula:*** taula a la que se li vol aplicar els permisos
* ***permís:*** tipus de permís que es vol aplicar
  + revoke: retirar tots els permisos
  + RWX: conjunció de qualsevol dels seus valor, èssent read/write/execution, respectivament
* ***grups:*** llistat de grups separats per coma, a qui se’ls aplica els permisos

# Auditoria

## Configurant Navigator Audit Server

1. Entrar al Cloudera Manager Admin Console.
2. Seleccionar Clusters > Cloudera Management Service.
3. Clicar la pestanya Instances.
4. Clicar Add Role Instances.
5. Seleccionar el rol Navigator Audit Server i assignar-ho a un host apropiat del clúster.
6. Clicar Continue. La pàgina de configuració de la base de dades mostra:
   * navigator audit server
     + database: nav
     + user: nav
     + pssword: xxxxx
   * navigator metadata server
     + database: navms
     + user: navms
     + pssword: xxxxx
7. Clicar Test Connection per verificar la comunicació entre el clúster i la base de dades externa.
8. Clicar Continue.
9. La pàgina Cluster Setup Review Changes es mostra.
10. Clicar Finish.
11. Iniciar els dos nous rols.
12. Després de reiniciar els servidors, a la pàgina principal, Impala pot mostrar alguns warnings.
13. Marcar l’opció de la configuració d'Impala, Enable Impala Audit Generation.

NOTA: el temps de vida de les dades emmagatzemades en el sistema d’auditoria ha estat establert en 365 dies, que és el màxim temps permès pel sistema.

# AnNexos

## Annex 1 (ajuda script create\_user.sh)

L'script create\_user.sh és un script que permet crear usuaris nous o afegir grups a un usuari ja existent.

L'script crearà l'usuari al KDC local, assignant-li una contrasenya. Després, en cada terminal del clúster de Cloudera, l'script crea l'usuari. També crearà els grups i se'ls assignarà a l'usuari. Tot seguit, generarà el directori de l'usuari a HDFS i sincronitzarà els usuaris del sistema amb els usuaris de HUE.

### Utilització amb paràmetres

Una de les formes en les que es pot utilitzar l'script és passant-hi la informació amb paràmetres.

Els paràmetres que accepta i que són sempre obligatoris són:

* **-u** el nom d'usuari que es crearà o al que s'hi afegiràn els grups
* **-p** la contrasenya que s'assiginarà al nou usuari. En el cas que l'usuari ja existeixi, aquest paràmetre s'ignora
* **-g** els grups a artefactes d'hbase que s'assignaran a l'usuari. Si s'afegeixen multiples grups, aquests han d'anar separats amb coma

# ./create\_user.sh -u ute -p ute123 -g hbase,hadoop,hue,talend

### Utilització amb fitxer .CSV

Una altra de les formes d'ús que accepta, és a través d'un fitxer en format .csv.

El paràmetre que s'ha de passar a l'script és el següent:

* **-f** la ruta al fitxer .csv on es troben els usuaris a carregar

Tot fitxer CSV que es carregui amb l'script ha de tenir les següents columnes:

* **Columna 0** el nom d'usuari que es crearà o al que s'hi afegiràn els grups
* **Columna 1** la contrasenya que s'assiginarà al nou usuari. En el cas que l'usuari ja existeixi, aquesta columna s'ignorarà
* **Columna 2-N** els grups que s'assignaràn a l'usuari

## Annex 2 (ajuda script hbase\_auth.sh)

L’escript ***hbase\_auth.sh*** s’invoca des de linia de comandes passat com a paràmetre el fitxer de configuració de permisos (vegeu apartat 6.7.4).

Per a cada linia del fitxer genera una linia de comanda del hbase shell I l’executa en finalitzar la seva lectura.

El mode de cridar l’script és:

# ./hbase\_auth.sh hbase\_auth.csv