Tot plugin Ranger

Control de versiones

Versión	Fecha de modificación	Responsable	Aprobador	Resumen de cambios
1.0	11/05/2023	Anjana Data Producto	Anjana Data Producto	Creación del documento

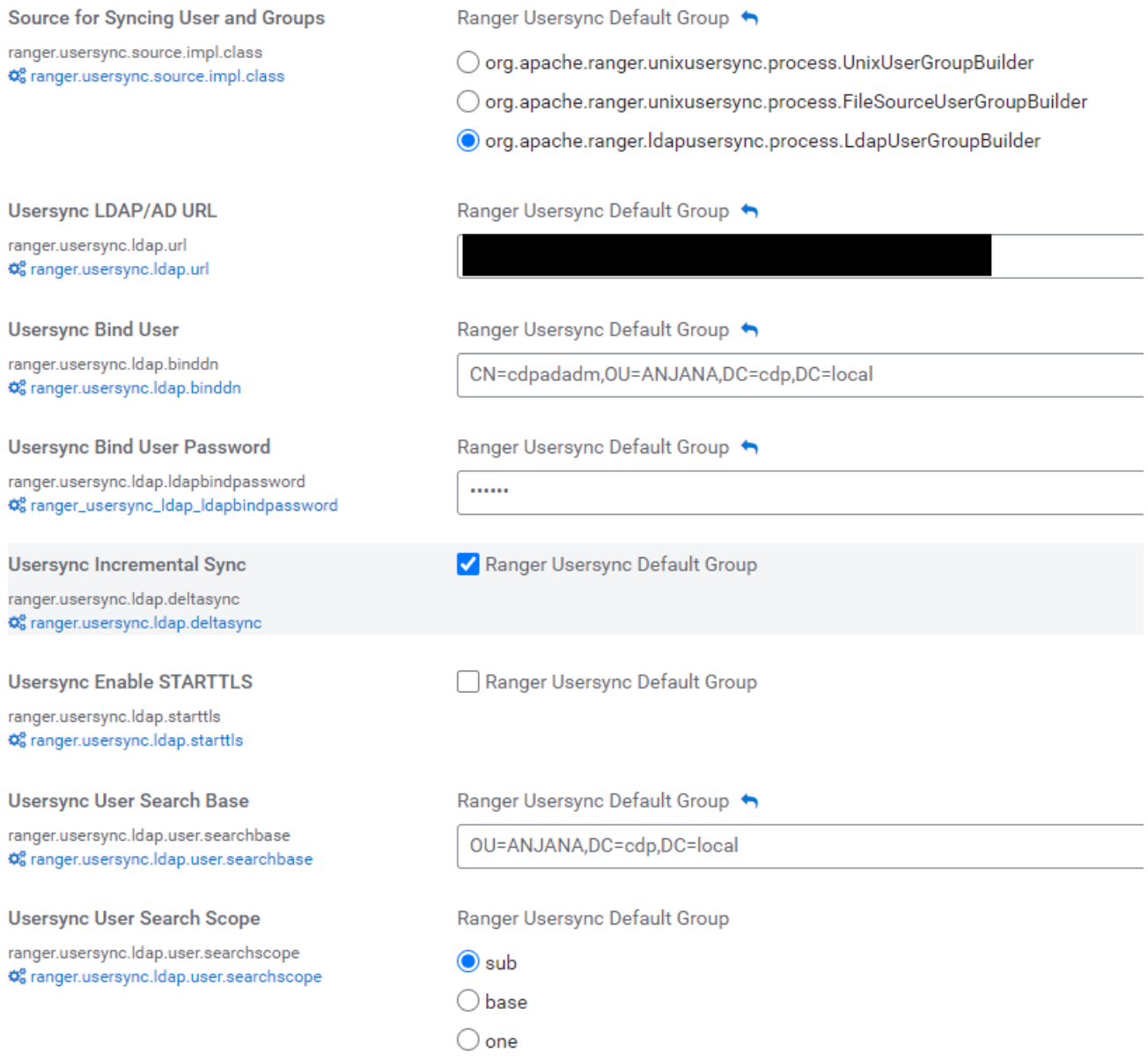
Modelo de integración

Gobierno activo

La gestión de acceso requiere el plugin “Tot plugin Ldap” para que genere los grupos que representan a los DSA e inserte o elimine usuarios a dichos grupos.

A su vez se requiere que Ranger sincronice dichos grupos para poder utilizarlos en sus políticas.

Las variables mostradas en las siguientes capturas son las propiedades que tendrán que adaptarse a la configuración del LDAP y Ranger que se tenga.



Usersync User Object Class

ranger.usersync.ldap.user.objectclass
ranger.usersync.ldap.user.objectclass

Ranger Usersync Default Group

user

Usersync User Search Filter

ranger.usersync.ldap.user.searchfilter
ranger.usersync.ldap.user.searchfilter

Ranger Usersync Default Group

Usersync User Name Attribute

ranger.usersync.ldap.user.nameattribute
ranger.usersync.ldap.user.nameattribute

Ranger Usersync Default Group

sAMAccountName

Usersync Referral

ranger.usersync.ldap.referral
ranger.usersync.ldap.referral

Ranger Usersync Default Group

- (•) ignore
- ( ) follow
- ( ) throw

Usersync Username Case Conversion

ranger.usersync.ldap.username.caseconversion
ranger.usersync.ldap.username.caseconversion

Ranger Usersync Default Group

- (•) none
- ( ) lower
- ( ) upper

Usersync Groupname Case Conversion

ranger.usersync.ldap.groupname.caseconversion
ranger.usersync.ldap.groupname.caseconversion

Ranger Usersync Default Group Deshacer

- (•) none
- ( ) lower
- ( ) upper

Usersync Enable User Search

ranger.usersync.user.searchenabled
ranger.usersync.user.searchenabled

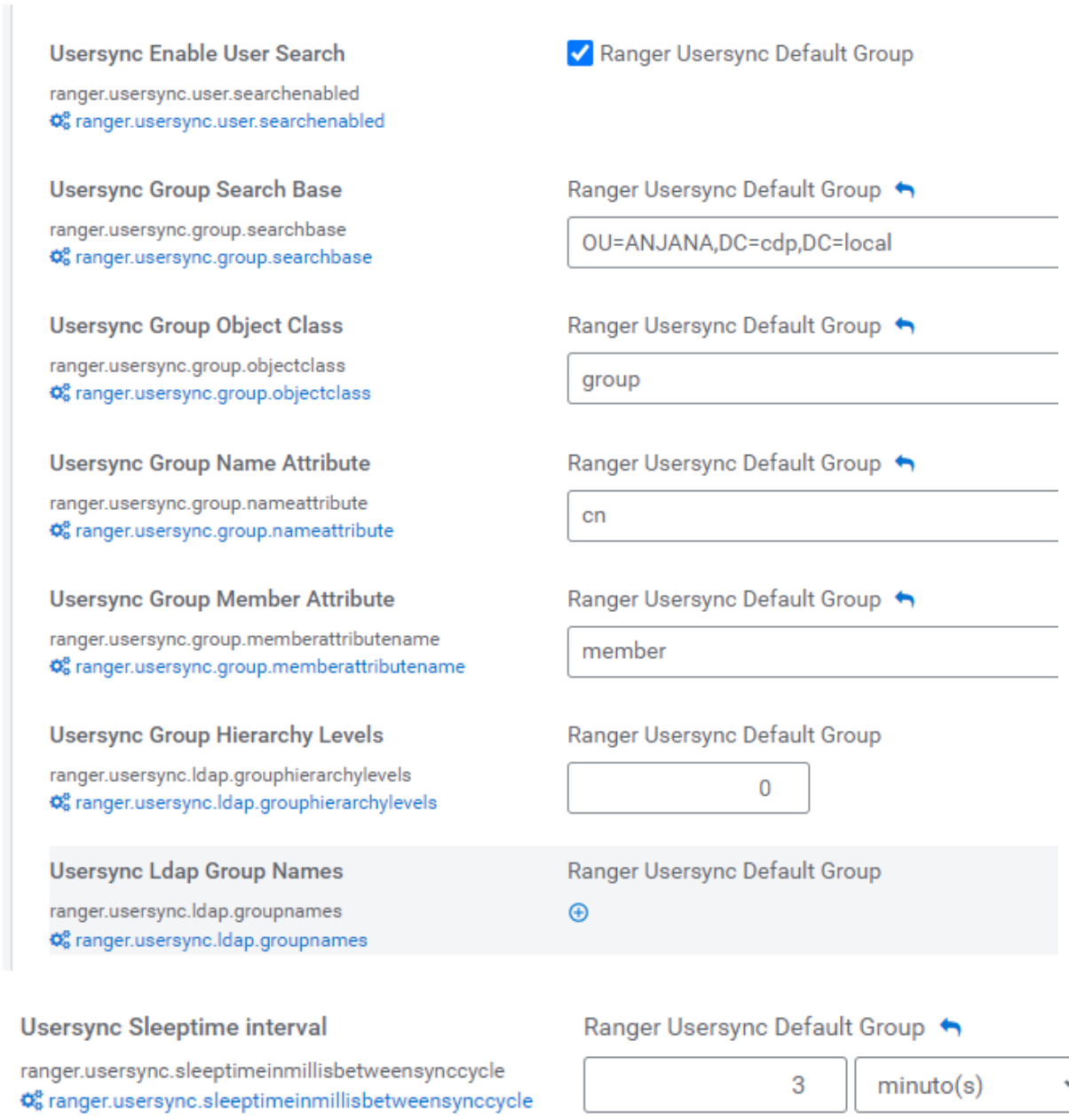
[x] Ranger Usersync Default Group

Usersync Group Search Base

ranger.usersync.group.searchbase
ranger.usersync.group.searchbase

Ranger Usersync Default Group

OU=ANJANA,DC=cdp,DC=local



El presente plugin creará o actualizará las políticas de HDFS y/o Hive para dar o quitar permisos a los grupos representados por DSAs.

Credenciales requeridas

Para permitir al plugin conectarse a Ranger se requieren dos cosas: un usuario de servicio con los permisos y credenciales correspondientes y certificados en la JVM para habilitar la conexión SSL a Ranger.

Generación y uso de los certificados

Para permitir la conexión entre el plugin y CDP es necesario realizar una serie de pasos previos para habilitar el SSL en el cluster y generar los certificados necesarios que el plugin requiere.

Proceso de activación de SSL/TLS con CDP

Actualmente con *Cloudera* es necesario activar *SSL/TLS* para poder hacer uso de *Ranger*. Si el asistente de este proceso falla hay muchas probabilidades de dejar el cluster completamente inoperable.

Preliminares

Durante la ejecución del asistente *Cloudera Manager* ha de conectarse, en nuestro caso con el usuario root, a todos los servidores del entorno, usando contraseña o bien usando claves pública/privada.

Para el par de claves público/privada hay que desplegar la clave pública de root a todos los nodos, además de habilitar el *root login* en la configuración del servidor *SSH*

```
Unset
sed -i 's/#PermitRootLogin prohibit-password/PermitRootLogin
yes/g' /etc/ssh/sshd_config

sed -i 's/PasswordAuthentication no/PasswordAuthentication
yes/g' /etc/ssh/sshd_config
```

Se recomienda realizar previamente un test de conexión manual desde el *Cloudera Manager* a todos los servidores del *Cluster.*

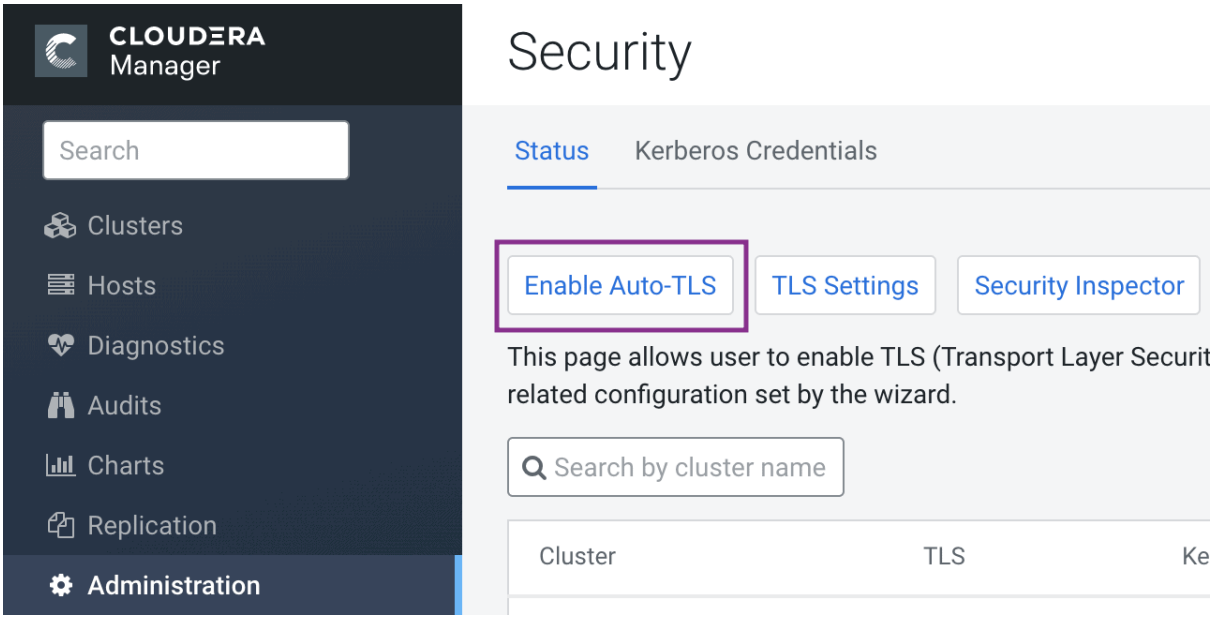
Asistente

El asistente consta de dos fases

- Generar la CA
- Pantalla de Resumen

Cuando se nos muestra la pantalla de resumen, sólo hemos de pulsar el botón de finalizar en la pantalla de resumen cuando finalice la operativa que nos describe que hagamos.

El asistente que se lanza desde al menú Administración > Seguridad > Enable TLS

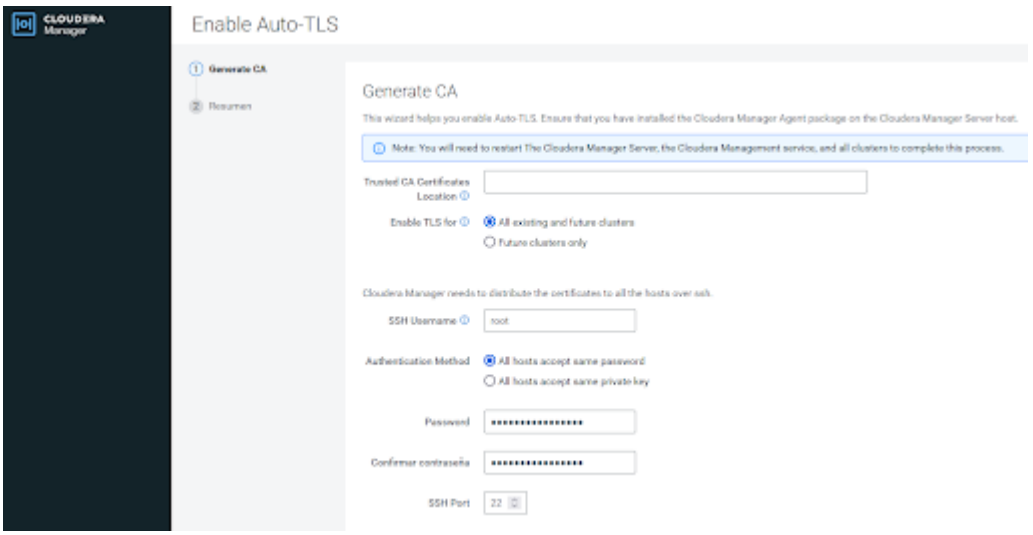


Asistente::Generar la CA

Esta primera pantalla del asistente nos permite o bien usar una *CA* que nos proporcionen o bien Cloudera genera una *CA* autofirmada.

En nuestro caso marcamos las opciones:

- All existing clusters and future clusters.
- All hosts accept the same password.
- ssh user name y password en nuestro caso root



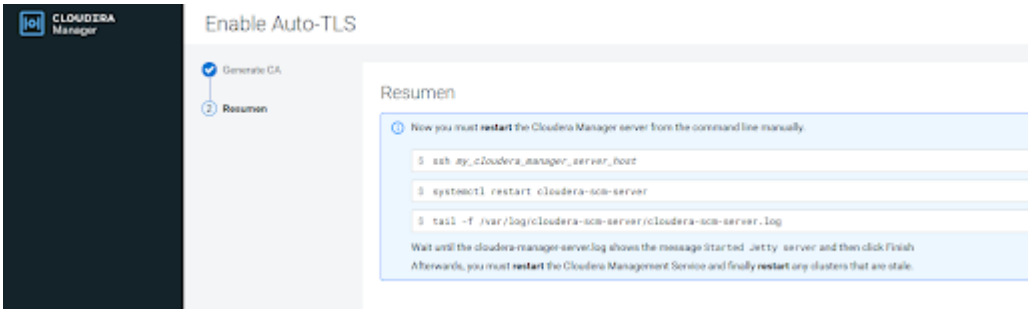
Asistente::Resumen

En esta pantalla NO debemos presionar finalizar hasta que se complete la operativa que nos muestra.

La operativa es:

1. conectarse a la máquina de *Cloudera Manager*
2. Reiniciar *Cloudera Manager*
3. Verificar en el log cuando se ha terminado de reiniciar

Sólo cuando esté completamente reiniciando el servido *Cloudera Manager* finalizaremos el asistente



PostInstalación

Después de la finalización del asistente tendremos que actualizar las configuraciones obsoletas de los clientes y aplicaciones.

TrustStores y Keystores

Cloudera almacena los JKS para certificados y claves en: /var/lib/cloudera-scm-agent/agent-cert

cm-auto-in_cluster_truststore.jks

cm-auto-in_cluster_ca_cert.pem

cm-auto-global_truststore.jks

cm-auto-global_cacerts.pem

cm-auto-host_key_cert_chain.pem

cm-auto-host_key.pw

cm-auto-host_key.pem

cm-auto-host_cert_chain.pem

cm-auto-host_keystore.jks

El password se puede sacar usando este comando:

```
Unset
grep -Eo "Djavax.net.ssl.trustStorePassword=[[:alnum:]]*"
$(find /run/cloudera-scm-agent/process/ -name "proc.json" |
grep HIVESERVER2 | sort | head -1) | sed
's/Djavax.net.ssl.trustStorePassword=//'
```

Conexión de SSL/TLS con CDP

El plugin utiliza una librería de Java para conectar por SSL a Ranger, es necesario que el plugin tenga acceso a los certificados generados anteriormente para conectar.

Existen dos maneras de hacerlo:

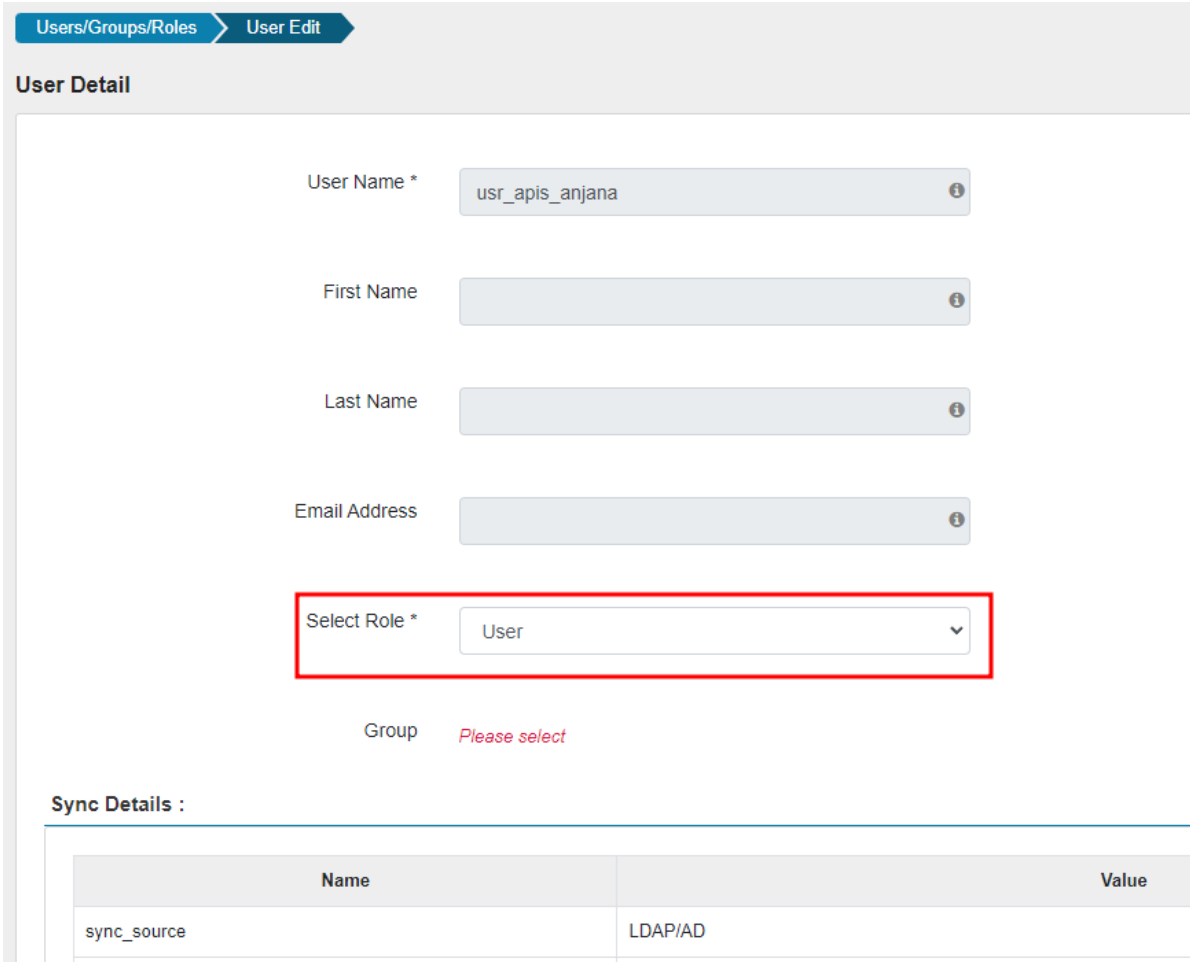
- Incluir el certificado en el comando de arranque del plugin incluyendo las siguientes variables de la JVM (deben):

  ```
  -Djavax.net.ssl.trustStore={ruta}\cm-auto-global_truststore.jks
  -Djavax.net.ssl.trustStorePassword=****
  ```

- Instalar los certificados en la JVM donde se ejecuta el plugin

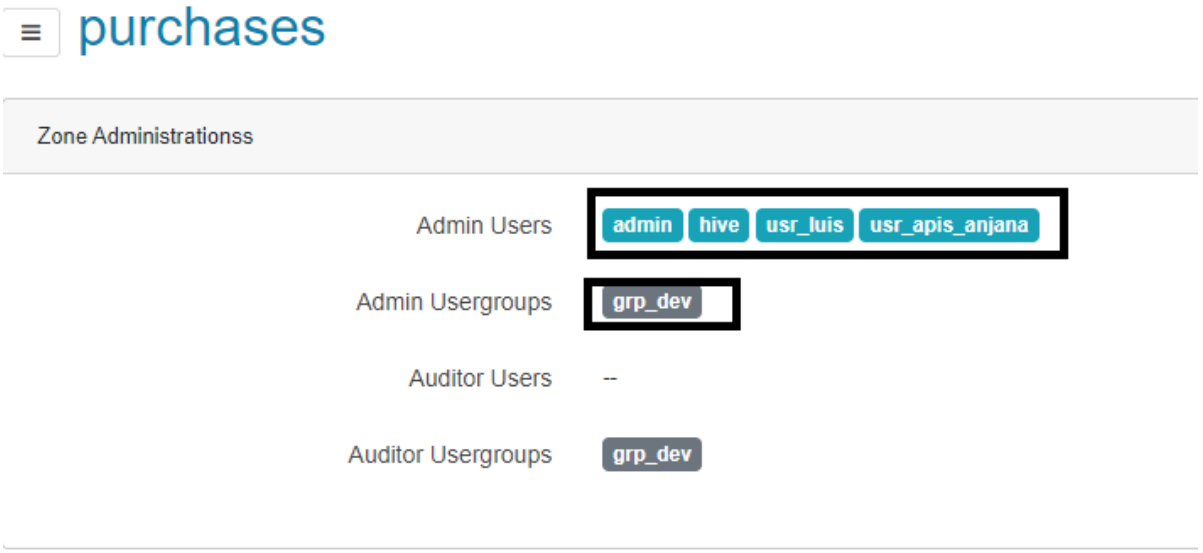
Credenciales del usuario

Se requiere la creación de un usuario de servicio de tipo User, ya sea creado desde Ranger o sincronizado desde LDAP.



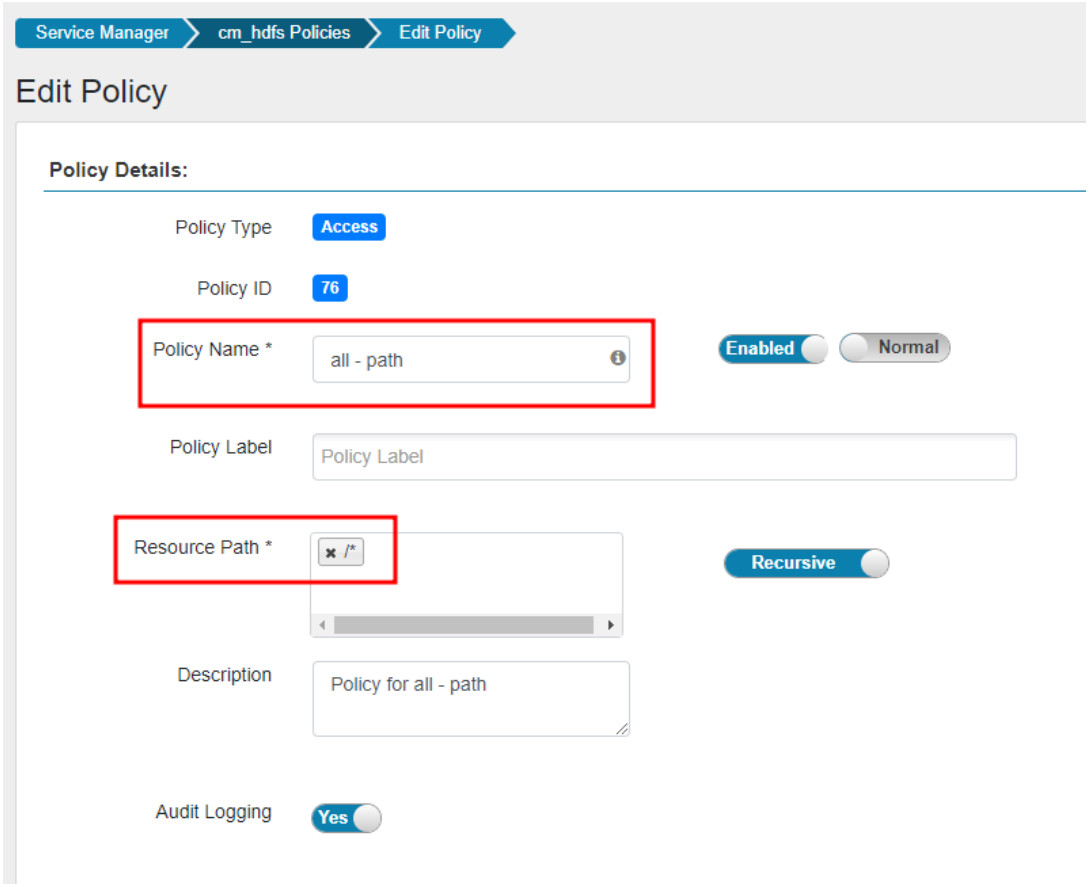
Dicho usuario necesita estar presente en ciertas políticas de HDFS y Hive que le permitan modificar todos los elementos que se quieren gobernar.

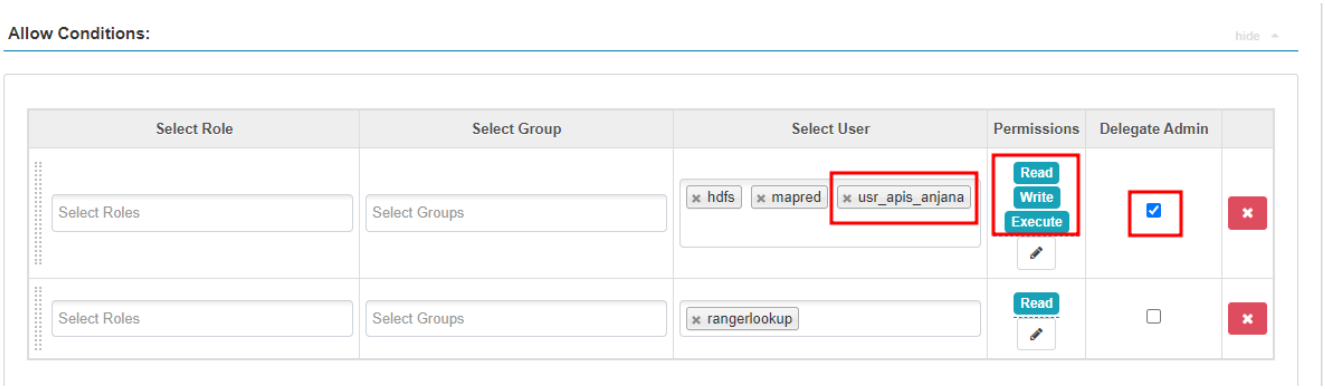
Si se quiere que las políticas estén en una zona de seguridad en particular (configurable) es necesario que el usuario de servicio sea administrador de la zona de seguridad, ya sea nominal o por pertenencia a un grupo.



HDFS

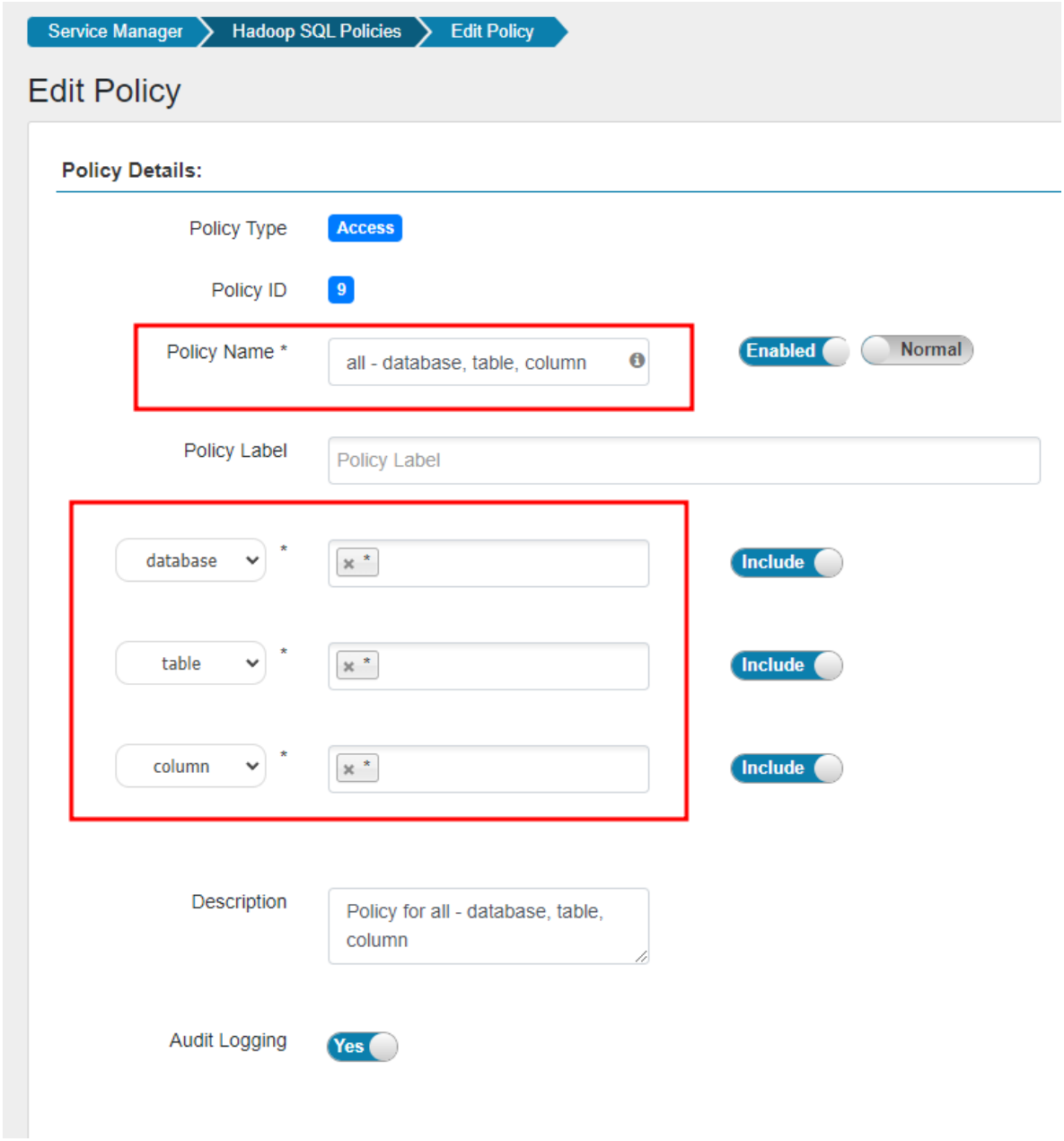
En la política definida por defecto “all - path” creada por Ranger en HDFS añadir al usuario que utilizará Anjana otorgándoles todos los permisos y marcando la opción “Delegate Admin”

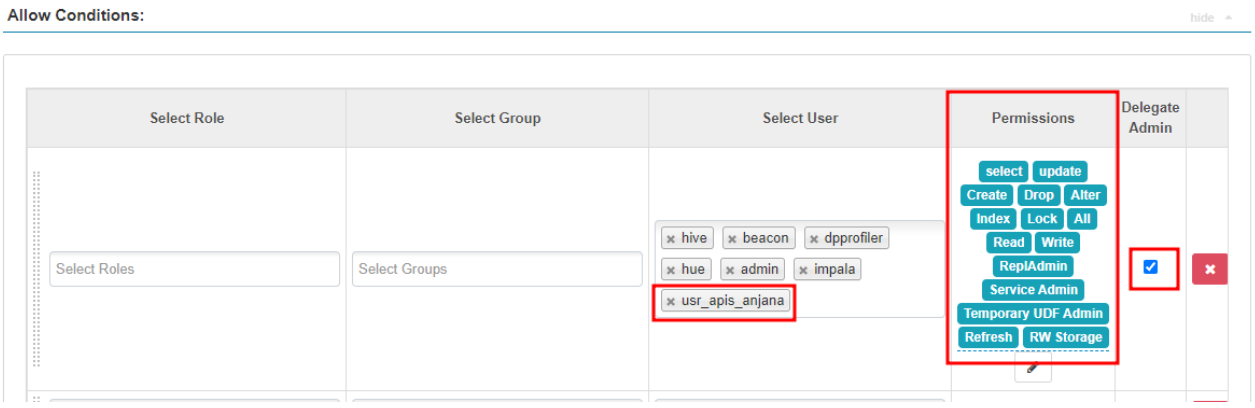




Hive

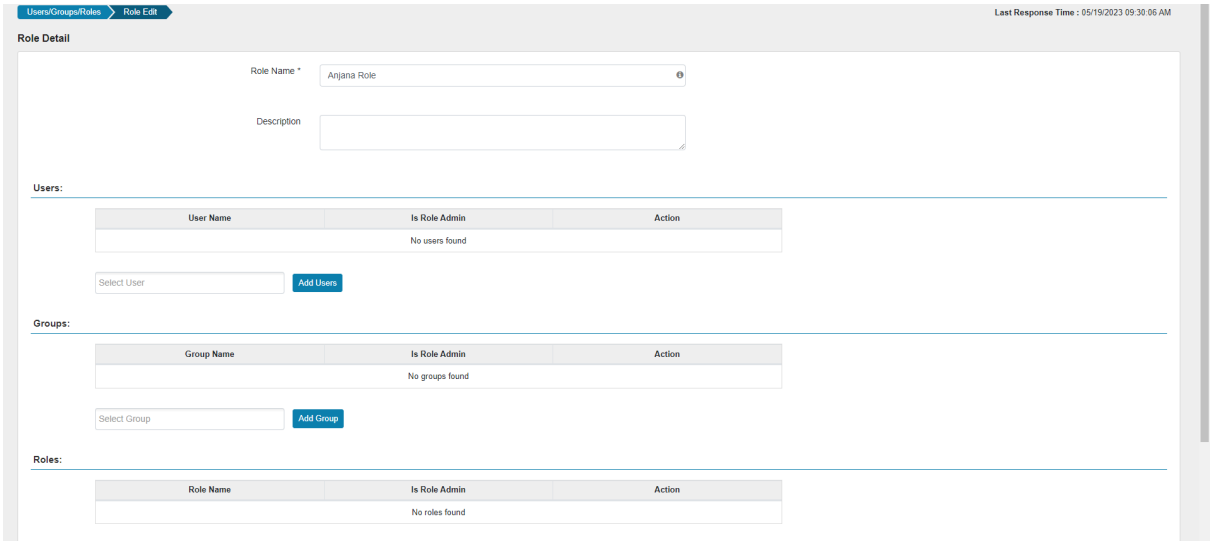
De manera similar a HDFS en la política definida por defecto “all - database, table, column” de Hadoop SQL añadir al usuario que utilizará Anjana otorgándoles todos los permisos y marcando la opción “Delegate Admin”.


Service Manager
Hadoop SQL Policies
Edit Policy
Edit Policy
Policy Details:
Policy Type
Access
Policy ID
9
Policy Name *
all - database, table, column
Enabled
Normal
Policy Label
Policy Label
database
table
column
Include
Include
Include
Description
Policy for all - database, table, column
Audit Logging
Yes


Allow Conditions:
hide
Select Role
Select Group
Select User
Permissions
Delegate Admin
Select Roles
Select Groups
hive
beacon
dpprofiler
hue
admin
impala
usr_apis_anjana
select
update
Create
Drop
Alter
Index
Lock
All
Read
Write
ReplAdmin
Service Admin
Temporary UDF Admin
Refresh
RW Storage

Rol de Anjana

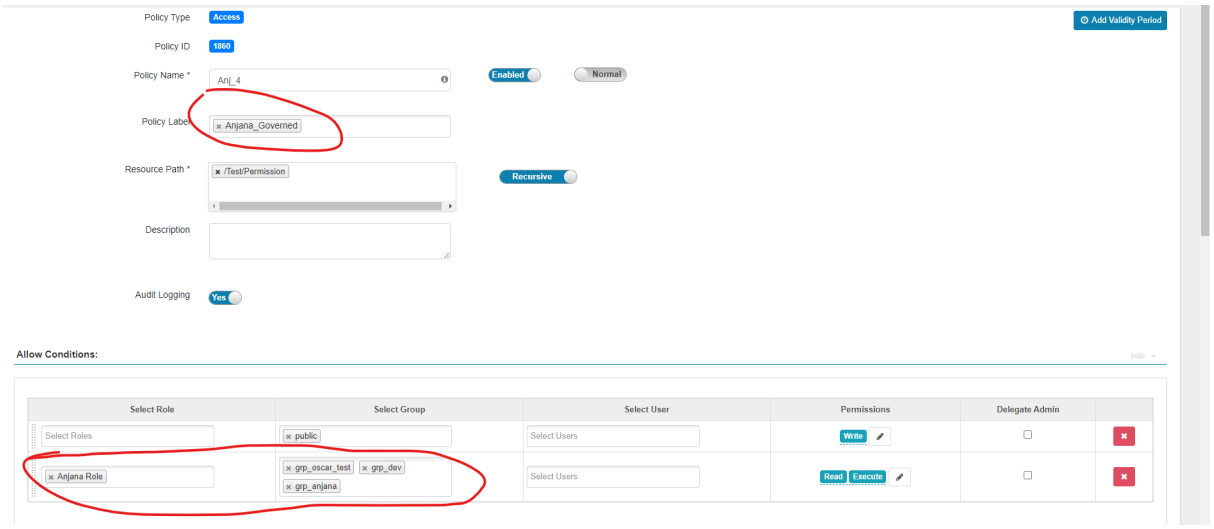
Para la agrupación de los permisos gobernados por Anjana es necesario crear un rol en Ranger que no contenga ningún usuario, grupo ni rol, a modo práctico funcionará como un tag en un registro de permisos. El nombre del rol debe ser Anjana Role.



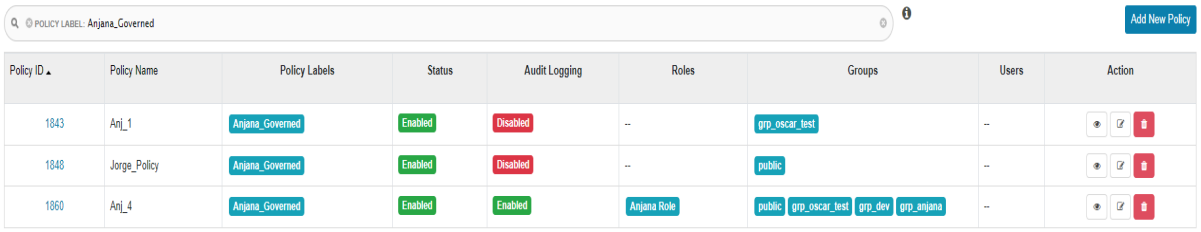
Limitaciones Ranger

Sólo es posible tener una política habilitada sobre un conjunto de objetos en particular (p.ej.: no es posible tener 2 políticas que actúan sobre el mismo fichero o tabla a la vez, pero sí una política que actúe sólo sobre el fichero y otra que actúe sobre el fichero y otro más), por ese motivo el plugin sólo actuará sobre políticas que apliquen a un solo recurso, ya sea un fichero/carpeta (HDFS) o una tabla (Hive).

Para facilitar saber qué políticas el plugin ha creado/modificado se le aplicará un tag a la política (Anjana Governed). Además de que todos los permisos gobernados por Anjana estarán agrupados en un único registro con el rol vacío creado anteriormente.



Con dicho tag se puede filtrar fácilmente qué políticas está interviniendo anjana y dentro de las mismas que permisos son los gobernados por el plugin, para que en caso de que se requiera modificar la política manualmente se puede identificar que fue creado por el plugin (y que no se debe de modificar).



Limitaciones Anjana

Por el procesamiento de los path a nivel de Hive, el path del objeto en Anjana que represente una tabla de Hive debe tener el path compuesto de una de estas dos maneras:

- catálogo u origen de datos/base de datos/tabla (EX: hive/db1/table1)
- base de datos/tabla (EX: db1/table1)

Siendo el carácter separador del path “/” configurable (mirar el ejemplo de configuración)

En los casos en los que se utilice el Active Directory y se utilice sAMAaccountName, por restricciones de dicha tecnología, el nombre de los grupos no debe ser mayor de 20 caracteres.

Acciones del plugin

Crear/Ampliar política

Esta acción se ejecuta cuando se aprueba un DSA en Anjana que contenga algún objeto gobernado con la tripleta configurada en el plugin. Esta acción se ejecutará con un retraso configurable en el plugin, porque hay que esperar a que Ranger sincronice con el LDAP el grupo que se creó previamente en el plugin de Ldap.

Como se ha explicado en las limitaciones, primero se buscará si existe una política que actúa únicamente sobre el recurso que el objeto representa, en caso de no encontrarla se creará una. Sobre esta política se creará un nuevo registro de acceso sobre el grupo que el DSA representa y le dará los permisos correspondientes (read/execute en HDFS y select en Hive), además de incluir el tag Anjana Governed para visibilidad.

Añadir que toda política de esta acción estará automáticamente habilitada. En caso de que existiera una sobre el recurso deshabilitada, se limpiará de todo permiso o exclusiones previos que tuviera antes de tratar sobre ella.

En el caso de políticas de HDFS, si la política que apunta al recurso no es recursiva, el plugin la pasará a recursiva quedando del mismo modo que si la hubiera creado el plugin.

Eliminar/reducir política

Esta acción se ejecuta cuando se expira un DSA en Anjana que contenga algún objeto gobernado con la tripleta configurada en el plugin o cuando se expira el objeto gobernado y está presente en algún DSA. Sobre la política asociada al recurso se eliminarán los registros de acceso.

En los casos de que todavía queden acceso gobernados por Anjana y se encuentra la política deshabilitada, se volverá a habilitar.

En los casos en los que al eliminar el acceso de Anjana todavía queden accesos gestionados por el cliente se eliminará el tag de Anjana Governed para indicar que Anjana ya no gobierna sobre dicho recurso y en el caso de que fueran los únicos registros existentes se eliminará la política también.

Ejemplo de configuración

```
server:
 port: 15021

totplugin:
```

```
location: http://totpluginrangerserver:15021/plugin/ranger/api/v1
server:
  url: http://totserver:15000/tot/
aris:
  - ari: "anja:totplugin:im:/cloudera/hive/devQA/"
    imAri: "anja:totplugin:im:/ldap/ldap/ldap/"
  - ari: "anja:totplugin:im:/cloudera/hdfs/devQA/"
    imAri: "anja:totplugin:im:/ldap/ldap/ldap/"
  - ari: "anja:totplugin:im:/cloudera/ranger/devQA/"
    imAri: "anja:totplugin:im:/ldap/ldap/ldap/"
connection:
  urlBaseRanger: "https://ip-10-150-100-136.eu-central-1.compute.internal:6182"
  userRanger: "usr_apis_anjana"
  pwdRanger: ******
delayInSeconds: 0
securityZone: "sales"
hive:
  hiveService: cm_hive
  valueTechnologyHive: hive
  hiveAuditLogin: true
  replaceHiveClientPolicyByAnjanaPolicyName: true
  pathSplit: "/"
hdfs:
  hdfsService: cm_hdfs
  valueTechnologyHdfs: hdfs
  hdfsAuditLogin: true
  replaceHdfsClientPolicyByAnjanaPolicyName: true
```

Se han de revisar las configuraciones comunes en el doc de configuraciones Anjana Data 4.4 - DS - Configuración técnica de Portal y microservicios

Configuraciones específicas:

- Connection:
    - urlBaseRange: Url de acceso a Ranger
    - userRanger: Usuario de servicio del plugin
    - pwdRanger: Contraseña del usuario de servicio
- delayInSeconds: Delay en segundos, usado en la aprobación del DSA, es recomendado ponerlo a un poco más que el tiempo de sincronización con LDAP configurado Ranger. Su valor predeterminado es 0.
- securityZone: Campo opcional, si relleno indica la zona de seguridad donde se van a crear las políticas.
- hive:
    - hiveService: Nombre del servicio de Hive dentro de Ranger sobre el que se quiere aplicar las políticas.

- valueTechnlogyHive: El valor en el campo technology que deben de tener los objetos gobernados para ser identificados como Hive y creadas sus políticas en su servicio.
- hiveAuditLogin: Campo opcional, indica si se quiere que las políticas creadas o actualizadas de hive por el plugin se auditen. Es false por defecto.
- replaceHiveClientPolicyByAnjanaPolicyName: Campo opcional, indica si se quiere permitir que el plugin modifique el nombre de las políticas de hive que actualiza con el nombre Anj_<Id del objeto que representa la tabla>. Es false por defecto.
- pathSplit: Campo opcional, el carácter por el cual se procesa el path de un objeto de hive para obtener su base de datos y que tabla representa. (EX: para database/table seria el caracter /).

- hdfs:
  - hdfsService: Nombre del servicio de HDFS dentro de Ranger sobre el que se quiere aplicar las políticas.
  - valueTechnlogyHdfs: El valor en el campo technology que deben de tener los objetos gobernados para ser identificados como HDFS y creadas sus políticas en su servicio.
  - hdfsAuditLogin: Campo opcional, indica si se quiere que las políticas creadas o actualizadas de HDFS por el plugin se auditen. Es false por defecto.
  - replaceHdfsClientPolicyByAnjanaPolicyName: Campo opcional, indica si se quiere permitir que el plugin modifique el nombre de las políticas de hdfs que actualiza con el nombre Anj_<Id del objeto que representa el/los fichero/s>. Es false por defecto.