



INRAE



Cati Sicpa

Ce document est mise à disposition selon les termes de la
[Licence Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/)

Gestion des nodes dans un cluster Apache Cassandra

28 août 2018

1 Objectifs

Ce document présente quelques tâches administratives à réaliser par moment pour maintenir un cluster Cassandra en bon état de fonctionnement. Les tâches administratives que nous considérons sont celles de la gestion des nodes dans un cluster Cassandra. Plus précisément, nous commencerons par décrire une procédure permettant d'ajouter un nouveau node à un cluster Cassandra. Ensuite, nous présenterons une démarche à suivre pour supprimer un node d'un cluster Cassandra. Nous donnerons enfin quelques consignes pour la réintroduction d'un node supprimé (qui fonctionne à nouveau) dans un cluster Cassandra.

2 Ajout d'un nouveau node dans un cluster

Lorsque qu'on se rapproche de la capacité maximale de stockage d'un cluster Apache Cassandra, les temps de réponses des requêtes en lecture et en écriture deviennent de plus en plus longues. Pour remédier à cette situation, il est nécessaire d'agrandir la taille du cluster en lui ajoutant quelques nodes en plus. Dans un cluster Cassandra, pour garantir une répartition équilibrée de la charge, il est recommandé que les différents racks d'un même data center aient le même nombre de nodes. Par conséquent, il faut ajouter le nombre de node nécessaire par data center pour que cette prescription soit respectée. L'ajout d'un nouveau node dans un cluster Cassandra peut se faire en suivant les 5 étapes présentées ci-dessous.

2.1 Installation de Apache Cassandra dans le nouveau node

Notre document intitulé "*Installation de Apache Cassandra sous CentOS 7*" décrit une procédure qui peut être suivie pour installer Cassandra dans un ordinateur utilisant le système d'exploitation CentOS 7.

2.2 Configuration des propriétés de Cassandra dans le nouveau node

La procédure décrite dans la Section 3 de notre document intitulé "*Configuration d'un cluster Apache Cassandra sous CentOS 7*" peut être suivie pour configurer les propriétés de Cassandra dans le nouveau node devant rejoindre notre cluster Cassandra.

2.3 Configuration de la sécurité dans le nouveau node

On peut suivre la procédure décrite dans la Section 2.2 de notre document intitulé *"Sécurisation d'un cluster Apache Cassandra"* pour sécuriser Cassandra dans le nouveau node.

2.4 Démarrage et vérification de Cassandra dans le nouveau node

On démarre Cassandra dans le nouveau node avec la commande suivante :

```
# sudo service Cassandra start
```

On vérifie ensuite que ce nouveau node a bien rejoint le cluster en exécutant la commande suivante :

```
# nodetool status
```

2.5 Suppression des données en trop dans tous les autres nodes

Après l'intégration du nouveau node dans le cluster Cassandra, les anciens nodes vont lui envoyer une partie de leurs données tout en conservant une copie des données transférées. Pour supprimer ces données en trop dans le cluster, nous devons exécuter dans tous les anciens nodes la commande ci-dessous :

```
# nodetool cleanup
```

Il est important d'attendre que cette commande s'exécute complètement dans un node avant de passer à un autre node.

3 Suppression d'un node dans un cluster

Les démarches de suppression d'un node dans un cluster Cassandra diffèrent selon que le node que l'on souhaite supprimer est en bon état ou en mauvais état de fonctionnement. Pour connaître le status ou l'état de santé des différents nodes, nous pouvons exécuter la commande ci-dessous :

```
# nodetool status
```

Les nodes en bon état de fonctionnement auront le status "UN" (Up Normal) tandis que les nodes injoignables ou en mauvais état de fonctionnement auront le status "DN" (Down).

3.1 Suppression d'un node en bon état

Sur un node en bon état de fonctionnement que l'on veut supprimer du cluster Cassandra, on exécute la commande suivante :

```
# nodetool decommission
```

Cette commande aura pour effet de copier les données contenues dans ce node vers les autres nodes avant de le supprimer du cluster et d'en informer les nodes restants.

3.2 Suppression forcée d'un node en panne

Si un node est en mauvais état de fonctionnement, on exécute la commande ci-dessous dans un node en bon état de fonctionnement

```
# nodetool assassinate <IP-address>
```

Dans cette commande il faut remplacer "`<IP-address>`" par l'adresse IP du node à supprimer. On exécute ensuite dans tous les nodes en bon état de fonctionnement la commande ci-dessous tout en veillant à attendre que cette commande s'exécute complètement dans un node avant de passer à un autre node :

```
# nodetool repair
```

Cette commande aura pour but de répartir les données détenues par le node injoignable ou en panne dans les nodes en bon état à partir de leurs réplicats. Il est important de noter que si un node utilisé comme "*seed node*" est supprimé, il faut absolument mettre à jour la liste des "*seed nodes*" configurée dans le fichier "`/etc/cassandra/conf/cassandra.yaml`" de tous les nodes en bon états en y enlevant l'adresse IP du "*seed node*" supprimé et en y ajoutant l'adresse IP d'un autre node que désignera pour remplacer ce "*seed node*" supprimé.

4 Reintroduction d'un node supprimé dans un cluster

Nous précisons que les données sont toujours présents dans les nodes supprimés d'un cluster Cassandra. Ainsi, pour éviter tout conflit entre les version des données lors de la réintroduction d'un node supprimé (qui est à nouveau en bon état de fonctionnement) dans un cluster Cassandra, on peut suivre les quatre étapes ci-dessous.

4.1 Suppression des données

On supprime toutes les données que ce node contient avec les commandes suivantes :

```
# sudo rm -rf /var/lib/cassandra/data/*
# sudo rm -rf /var/lib/cassandra/commitlog/*
# sudo rm -rf /var/lib/cassandra/saved_caches/*
# sudo rm -rf /var/lib/cassandra/hints/*
```

4.2 Mise à jour des "*seed nodes*"

On met éventuellement à jour la liste des "*seed nodes*" en spécifiant comme valeur du paramètre "*seeds*" contenu dans le fichier "`/etc/cassandra/conf/cassandra.yaml`" les adresses IP de tous les "*seed nodes*" présents dans le cluster. Cette étape est nécessaire s'il y a eu entre temps une modification des "*seed nodes*" dans le cluster.

4.3 Choix de la position du node

On change éventuellement le positionnement du node dans le cluster en spécifiant le data center et le rack comme valeur des paramètres respectifs *"dc"* et *"rack"* contenus dans le fichier *"/etc/cassandra/conf/cassandra-rackdc.properties"*. Cette étape est nécessaire si on souhaite insérer ce node à un autre endroit du cluster.

4.4 Démarrage du node

On redémarre le node avec la commande suivante :

```
# sudo service cassandra start
```

Cette commande permettra au node de rejoindre le cluster exactement à l'emplacement indiqué. On vérifie que ce node fait parti à nouveau du cluster avec la commande suivante :

```
# sudo nodetool status
```

5 Conclusion

Dans ce document, nous avons montré comment accomplir certaines opérations de maintenance dans un cluster Cassandra en cours de fonctionnement. Plus précisément, nous avons décrit les procédures d'ajout et de suppression d'un node dans un cluster Cassandra. Dans le prochain travail, nous allons développer une application qui exploite un cluster Apache Spark pour transformer et structurer les données brutes avant de les intégrer dans un cluster Cassandra ou Solr.

Références