

MySQLCluster sur docker

Nous allons mettre en place un cluster MySQL basé sur MySQL Cluster.

Mise en place du réseau

```
docker network create cluster --subnet=192.168.0.0/16
```

Puis récupérons les fichiers ou recopier les fichier de configuration:

```
https://github.com/mysql/mysql-docker/tree/mysql-cluster
```

Le fichier my.cnf est ainsi:

```
# Copyright (c) 2017, Oracle and/or its affiliates. All rights reserved.
#
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; version 2 of the License.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

[mysqld]
ndbcluster
ndb-connectstring=192.168.0.2
user=mysql

[mysql_cluster]
ndb-connectstring=192.168.0.2
```

et le fichier mysqlcluster.cnf

```
# Copyright (c) 2017, Oracle and/or its affiliates. All rights reserved.
#
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; version 2 of the License.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
```

```
[ndbd default]
NoOfReplicas=2
DataMemory=80M
IndexMemory=18M
```

```
[ndb_mgmd]
NodeId=1
hostname=192.168.0.2
datadir=/var/lib/mysql
```

```
[ndbd]
NodeId=2
hostname=192.168.0.3
datadir=/var/lib/mysql
```

```
[ndbd]
NodeId=3
hostname=192.168.0.4
datadir=/var/lib/mysql
```

```
[mysqld]
NodeId=4
hostname=192.168.0.10
```

Lancement du noeud de management

```
docker run -d --net=cluster --name=management1 --ip=192.168.0.2 mysql/mysql-cluster ndb_mgmd
```

Puis lancons les noeud de données:

```
docker run -d --net=cluster --name=ndb1 --ip=192.168.0.3 mysql/mysql-cluster ndbd
```

```
docker run -d --net=cluster --name=ndb2 --ip=192.168.0.4 mysql/mysql-cluster ndbd
```

Et enfin on peut executer le noeud SQL:

```
docker run -d --net=cluster --name=mysql1 --ip=192.168.0.10 -e MYSQL_RANDOM_ROOT_PASSWORD=true  
mysql/mysql-cluster mysqld
```

Ce noeud va générer un password aléatoire :

```
docker logs mysql1 2>&1 | grep PASSWORD  
[Entrypoint] GENERATED ROOT PASSWORD: =EbISQomAxv0mnam4d9EJigIjwA
```

Changement du mot de passe:

```
docker exec -it mysql1 mysql -uroot -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 8  
Server version: 5.7.30-ndb-7.6.14-cluster-gpl MySQL Cluster Community Server (GPL)  
  
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'MyNewPass';  
Query OK, 0 rows affected (0.06 sec)
```

Le management du cluster se fait via :

```
docker run -it --net=cluster mysql/mysql-cluster ndb_mgm
```

La modification du fichier mysqlcluster.cnf permet de rajouter un noeud de donnée et de relancer

```
docker run -d --net=cluster --name=management1 --ip=192.168.0.2 -v
/home/pilou/Formation/cluster/mysql-cluster.cnf:/etc/mysql-cluster.cnf mysql/mysql-cluster
ndb_mgmd
9ac595c65ddf47568f20fa835e10d45d0e7adcaf40bab818ac94b9afd20524ec
pilou@pilou-pc: ~/Formation/cluster$ docker logs management1[Entrypoint] MySQL Docker Image
7.6.14-1.1.16-cluster
[Entrypoint] Starting ndb_mgmd
MySQL Cluster Management Server mysql-5.7.30 ndb-7.6.14
2020-06-09 20:25:07 [MgmtSrvr] INFO      -- The default config directory '/usr/mysql-cluster'
does not exist. Trying to create it...
2020-06-09 20:25:07 [MgmtSrvr] INFO      -- Successfully created config directory
2020-06-09 20:25:07 [MgmtSrvr] WARNING  -- at line 19: [DB] IndexMemory is deprecated, will
use Number bytes on each ndbd(DB) node allocated for storing indexes instead
2020-06-09 20:25:07 [MgmtSrvr] INFO      -- Got initial configuration from '/etc/mysql-
cluster.cnf', will try to set it when all ndb_mgmd(s) started
2020-06-09 20:25:07 [MgmtSrvr] INFO      -- Node 1: Node 1 Connected
2020-06-09 20:25:07 [MgmtSrvr] INFO      -- Id: 1, Command port: *:1186
==INITIAL==
2020-06-09 20:25:07 [MgmtSrvr] INFO      -- MySQL Cluster Management Server mysql-5.7.30 ndb-
7.6.14 started
2020-06-09 20:25:08 [MgmtSrvr] INFO      -- Node 1 connected
2020-06-09 20:25:08 [MgmtSrvr] INFO      -- Starting initial configuration change
2020-06-09 20:25:08 [MgmtSrvr] INFO      -- Configuration 1 committed
2020-06-09 20:25:08 [MgmtSrvr] INFO      -- Config change completed! New generation: 1
==CONFIRMED==
```

Outil NDB

- ndbinfo_select_all - Sélection dans les tables ndbinfo
- ndbmtbd - Le démon de nœud de données de cluster NDB (multi-thread)
- ndb_mgmd - Le démon du serveur de gestion de cluster NDB
- ndb_mgm - Le client de gestion de cluster NDB

- `ndb_blob_tool` - Vérifie et répare les colonnes BLOB et TEXT des tables de cluster NDB
- `ndb_config` - Extraire les informations de configuration du cluster NDB
- `ndb_cpcd` - Tests automatiques pour le développement NDB
- `ndb_delete_all` - Supprime toutes les lignes d'une table NDB
- `ndb_desc` - Décrit les tables NDB
- `ndb_drop_index` - Supprime l'index d'une table NDB
- `ndb_drop_table` - Supprime une table NDB
- `ndb_error_reporter` - Utilitaire de rapport d'erreurs NDB
- `ndb_import` - Importer des données CSV dans le NDB

Sizing

Executer

```
perl ./bin/ndb_size.pl --socket /tmp/mysql.sock --user=root --password=piloupilou </source>
```

Parameter	Default	4.1	5.0	5.1
NoOfAttributes		1000	3676*	3676*
NoOfUniqueHashIndexes		64	0	0
IndexMemory (KB)		18432	4592	3568
NoOfOrderedIndexes		128	233*	233*
DataMemory (KB)		81920	55328	55328
NoOfTriggers		768	1985*	1985*
NoOfTables		128	438*	438*

Fichier de configuration conseillé:

LE fichier suivant est un fichier config.ini correct pour la plupart des cluster:

```
[tcp default] SendBufferMemory=2M ReceiveBufferMemory=2M
```

```
[ndb_mgmd default] datadir=X
```

```
[ndbd default]
```

1. redundancy:

```
NoOfReplicas=2
```

1. avoid swapping:

```
LockPagesInMainMemory=1
```

1. Bypass FS cache (you should test if this works for you or not)

Odirect=1

1. DataMemory (memory for records and ordered indexes)

DataMemory=2048M

1. IndexMemory (memory for Primary key hash index and unique hash index)
2. Usually between 1/6 or 1/8 of the DataMemory is enough, but depends on the
3. number of unique hash indexes (UNIQUE in table def)

IndexMemory=256M

1. Redolog
2. size of each redo log fragment, 4 redo log fragment makes up on fragment log file.
3. A bigger Fragment log file size than the default 16M works better with high write load
4. and is strongly recommended!!

FragmentLogFileSize=256M

1. Set NoOfFragmentLogFiles to $6 \times \text{DataMemory} [\text{in MB}] / (4 * \text{FragmentLogFileSize} [\text{in MB}])$
2. Thus, $\text{NoOfFragmentLogFiles} = 6 * 2048 / 1024 = 12$
3. The "6xDataMemory" is a good heuristic and is STRONGLY recommended.

NoOfFragmentLogFiles=12

1. RedoBuffer of 32M should let you restore/provisioning quite a lot of data in parallel.
2. If you still have problems ("out of redobuffer"), then you probably have to slow disks and
3. increasing this will not help, but only postpone the inevitable.

RedoBuffer=32M

1. table related things

MaxNoOfTables=4096 MaxNoOfAttributes=24756 MaxNoOfOrderedIndexes=2048

MaxNoOfUniqueHashIndexes=512

1. Operation records
2. MaxNoOfConcurrentOperations=100000 means that you can load any mysqldump file into cluster.

MaxNoOfConcurrentOperations=100000

1. Checkpointing...

Diskcheckpointspeed=10M Diskcheckpointspeedinrestart=100M
TimeBetweenGlobalCheckpoints=1000

1. the default value for TimeBetweenLocalCheckpoints is very good

TimeBetweenLocalCheckpoints=20

1. Realtime extensions (only in MySQL Cluster 6.3 (CGE 6.3) , read this how to use this)
2. SchedulerSpinTimer=400
3. SchedulerExecutionTimer=100
4. RealTimeScheduler=1
5. LockMaintThreadsToCPU=[cpuid]
6. LockExecuteThreadToCPU=[cpuid]

1. If you use MySQL Cluster 6.3 (CGE 6.3) and are tight on disk space, e.g ATCA.
2. You should also then lock cpu's to a particular core.
3. CompressedLCP=1
4. CompressedBackup=1

datadir=X

[ndb_mgmd] hostname=...

1. second management server for redundancy
2. [ndb_mgmd]
3. hostname=...

[ndbd] hostname=...

[ndbd] hostname=...

[mysqld]

...

[mysqld]

Revision #7

Created 9 June 2020 17:25:08 by ggpilou2

Updated 22 November 2021 21:24:16 by ggpilou2