

# Mysql sur un cluster de 3 machines

## I. Les prerequis

D'abord nous devons avoir 3 machines . Elles sont représentées dans le tableau ci-dessous

Machine	Adresse IP
UBUNTU 22.04_SRV-DEB-1	192.168.64.2
Macdeidy_SRV-DEB-2	192.168.1.5
AmadouDELL_SRV-DEB-3	192.168.1.3

Mon objectif étant d'héberger sur ce cluster la **base de données d'un site**. A partir de mon serveur **UBUNTU 22.04\_SRV-DEB-1**, ou se situe actuellement ma base de données, je vais me connecter a l'instance MariaDB :

```
root@ubuntudeidrissa:/# mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show variables like 'default_storage_engine';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| default_storage_engine | InnoDB |
+-----+-----+
1 row in set (0,026 sec)

MySQL [(none)]>
```

Le moteur de recherche est innobdb

## II. Installation de MariaDB et Galera 4 sur tous les noeuds

### 1) Installation sur le serveur **UBUNTU 22.04\_SRV-DEB-1**

```
root@ubuntudeidrissa:/# sudo apt-key adv --fetch-keys 'https://mariadb.org/mariadb_release_signing_key.asc'
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
Executing: /tmp/apt-key-gpghome.weBvFvFXxj/gpg.1.sh --fetch-keys https://mariadb.org/mariadb_release_signing_key.asc
gpg: requesting key from 'https://mariadb.org/mariadb_release_signing_key.asc'
gpg: key F1656F24C74CD1D8: public key "MariaDB Signing Key <signing-key@mariadb.org>" imported
gpg: Total number processed: 1
gpg:      imported: 1
Atteint :5 http://ports.ubuntu.com/ubuntu-ports jammy-backports InRelease
Réception de :6 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease [110 kB]
Réception de :7 http://ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 Packages [1 311 kB]
Réception de :8 http://ports.ubuntu.com/ubuntu-ports jammy-updates/universe arm64 Packages [1 001 kB]
Ign :9 http://mirror.nodesdirect.com/mariadb/repo/10.6/ubuntu focal InRelease
Ign :9 http://mirror.nodesdirect.com/mariadb/repo/10.6/ubuntu focal InRelease
Ign :9 http://mirror.nodesdirect.com/mariadb/repo/10.6/ubuntu focal InRelease
Err :9 http://mirror.nodesdirect.com/mariadb/repo/10.6/ubuntu focal InRelease
  Connexion à mirror.nodesdirect.com: 80 (23.92.92.94) impossible, délai de connexion dépassé
Reading package lists... Done
E: The repository 'file:/cdrom jammy Release' no longer has a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
root@ubuntudeidrissa:/#
Réception de :4 http://ports.ubuntu.com/ubuntu-ports jammy-updates InRelease [119 kB]
```

Intallation de MariaDB Galera Cluster réussie

## - Installation de mariadb server et de galera4

```
idrisa@ubuntu:~$ sudo apt-get install mariadb-server
[sudo] password for idrisa:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
mariadb-server est déjà la version la plus récente (1:10.6.16-0ubuntu0.22.04.1)
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libevent-pthreads-2.1-7 libmecab2 libprotobuf-lite23 mecab-ipadic
  mecab-ipadic-utf8 mecab-utils
Veuillez utiliser « sudo apt autoremove » pour les supprimer.
0 mis à jour, 0 nouvellement installés, 0 à enlever et 122 non mis à jour.
idrisa@ubuntu:~$
```

```
root@ubuntu:~# sudo apt install mariadb-server galera-4 mariadb-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
```

## 2) Installation sur machine2

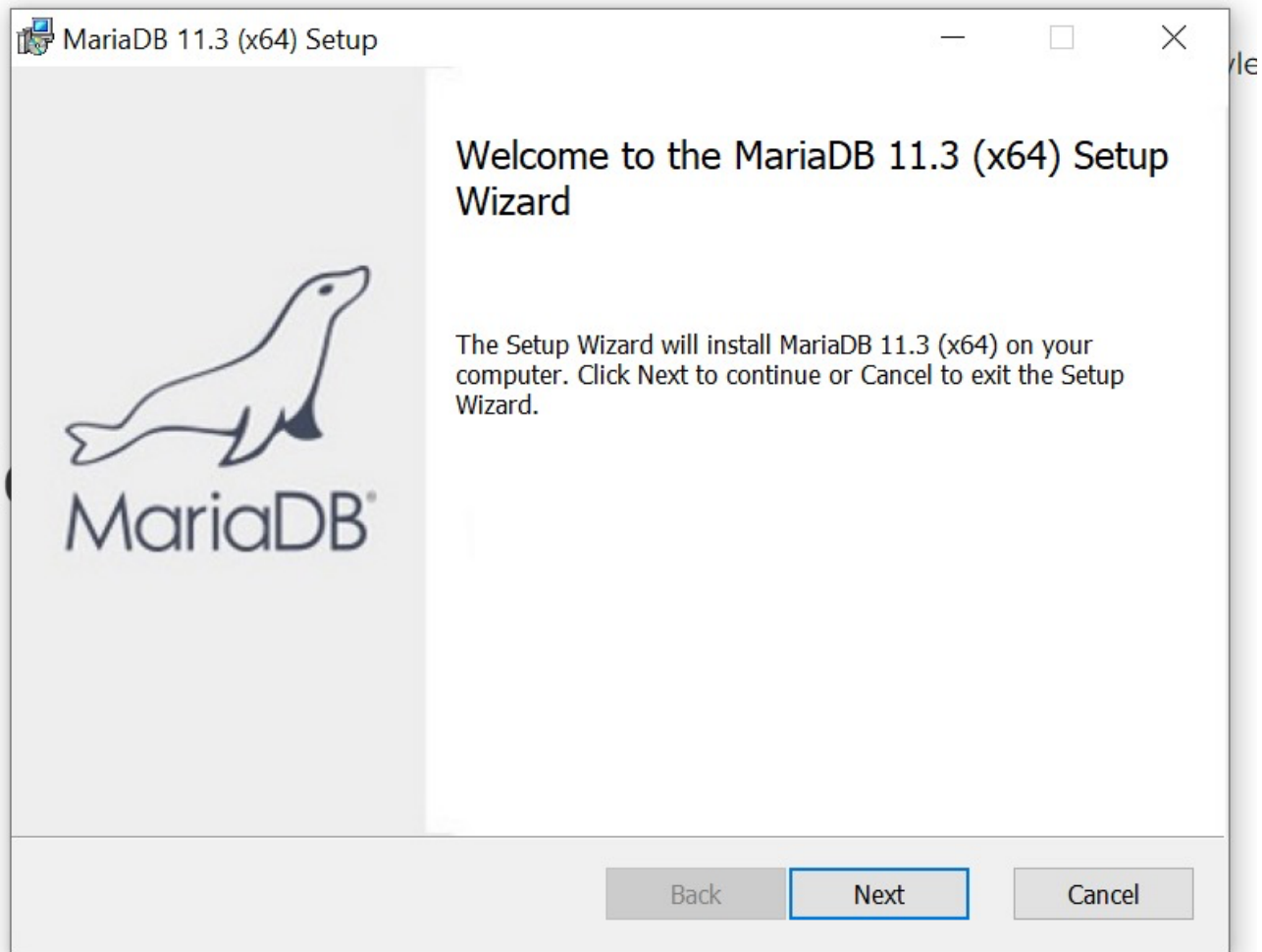
```
idrisa@MacBook-Pro-de-Idrisa: ~ % brew update
=> Updating Homebrew...
Already up-to-date.
idrisa@MacBook-Pro-de-Idrisa: ~ % brew install mariadb

=> Downloading https://ghcr.io/v2/homebrew/core/mariadb/manifests/11.3.2
===== 100.0%
=> Fetching dependencies for mariadb: mecab, mecab-ipadic, msgpack, openssl@3 and groonga
=> Downloading https://ghcr.io/v2/homebrew/core/mecab/manifests/0.996-3
===== 100.0%
=> Fetching mecab
=> Downloading https://ghcr.io/v2/homebrew/core/mecab/blobs/sha256:492e9bdfcbb7968c83dfff7736cd94ba5ad97df0c164bb62c1fe9f9e389881ceb
===== 100.0%
=> Downloading https://ghcr.io/v2/homebrew/core/mecab-ipadic/manifests/2.7.0-20070801
===== 100.0%
=> Fetching mecab-ipadic
=> Downloading https://ghcr.io/v2/homebrew/core/mecab-ipadic/blobs/sha256:fd42086389a7302de36628435004c2f8de2f55b01f6fd8b5a74529779fc2754a
===== 100.0%
=> Downloading https://ghcr.io/v2/homebrew/core/msgpack/manifests/6.0.1
===== 100.0%
=> Fetching msgpack
=> Downloading https://ghcr.io/v2/homebrew/core/msgpack/blobs/sha256:9666b5d9ac86ec85fdcf9195aa1064bf8604d9c283a44c217a74762f4bc0cd0c
===== 100.0%
=> Downloading https://ghcr.io/v2/homebrew/core/openssl@3/manifests/3.2.1-1
===== 100.0%
=> Fetching openssl@3
=> Downloading https://ghcr.io/v2/homebrew/core/openssl@3/blobs/sha256:020785e015f7b8ef638abc5835890bf3f0273c1e0c0a54b2f749e82cab0ddee
===== 100.0%
=> Downloading https://ghcr.io/v2/homebrew/core/groonga/manifests/14.0.2
===== 100.0%
=> Fetching dependencies for groonga: ca-certificates
=> Downloading https://ghcr.io/v2/homebrew/core/ca-certificates/manifests/2024-03-11
===== 100.0%
=> Fetching ca-certificates
=> Downloading https://ghcr.io/v2/homebrew/core/ca-certificates/blobs/sha256:cab828953672906e00a8f25db751977b8dc4115f021f8df82b644ade083dadb
===== 100.0%
=> Fetching groonga
=> Downloading https://ghcr.io/v2/homebrew/core/groonga/blobs/sha256:462ed9ac89447742d4d4d75cd21f91a62f5679dc6f2c3842732ca08a23f5a379
===== 100.0%
=> Fetching mariadb
=> Downloading https://ghcr.io/v2/homebrew/core/mariadb/blobs/sha256:a9f71222869b1e0f9cfe4cc0fd5ee340755efb5b31a26483a726b3a577a86828
===== 100.0%
Error: Cannot install mariadb because conflicting formulae are installed.
mysql: because mariadb, mysql, and percona install the same binaries

Please `brew unlink mysql` before continuing.

Unlinking removes a formula's symlinks from /opt/homebrew. You can
link the formula again after the install finishes. You can `--force` this
install, but the build may fail or cause obscure side effects in the
resulting software.
idrisa@MacBook-Pro-de-Idrisa: ~ %
```

### 3) installation sur la machine 3



## III. Configuration de MariaDB Galera Cluster

Nous allons déclarer la configuration de notre cluster sur le serveur **UBUNTU 22.04\_SRV-DEB-1**. Pour cela, nous devons modifier le fichier de configuration "60-galera.cnf" qui se situe dans le dossier `/etc/mysql/mariadb.conf.d/`. Vous pouvez le vérifier avec cette commande :

```
idrisa@ubuntuidrisa:~$ ls /etc/mysql/mariadb.conf.d/
50-client.cnf          50-mysqld_safe.cnf  60-galera.cnf
50-mysql-clients.cnf  50-server.cnf
idrisa@ubuntuidrisa:~$
```

Normalement, ce fichier est livré par défaut avec les versions récentes de MariaDB.

Editons et configurons ce fichier

```
GNU nano 6.2 /etc/mysql/mariadb.conf.d/60-galera.cnf *
[galera]
# Mandatory settings
wsrep_on = ON
wsrep_provider = /usr/lib/galera/libgalera_smm.so
wsrep_cluster_name = "Galera_Cluster_IT-Connect"
wsrep_cluster_address = gcomm://192.168.64.2,192.168.1.5,192.168.1.3
binlog_format = row
default_storage_engine = InnoDB
innodb_autoinc_lock_mode = 2
innodb_force_primary_key = 1

# Allow server to accept connections on all interfaces.
bind-address = 0.0.0.0

# Optional settings
#wsrep_slave_threads = 1
#innodb_flush_log_at_trx_commit = 0
log_error = /var/log/mysql/error-galera.log

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

Par défaut, dans les journaux, Galera Cluster va utiliser le nom d'hôte du serveur. Si l'on souhaite utiliser une autre valeur, dans ce cas, il faut définir cette option à laquelle on peut aussi ajouter l'adresse IP :

```
wsrep_node_name = "SRV-DEB-1"
wsrep_node_address = "192.168.64.2"
```

Si l'on veut parler un peu d'**optimisation des performances**, alors l'option "**wsrep\_applier\_threads**" est intéressante afin de jouer sur **le nombre de thread actifs pour traiter les opérations de réplication**. A ce sujet, il n'y a pas de formule magique, mais il faut que ce soit au moins

égal au nombre de coeurs de votre processeur, voire même le double ne me choque pas.

```
wsrep_applier_threads = 2
```

## IV. Démarrer le cluster Galera

Dans le but d'initialiser notre cluster Galera avec son noeud primaire, nous allons poursuivre sur le serveur SRV-DEB-1 qui contient notre configuration (60-galera.cnf) et la base de données a répliquer. Commençons par stopper MariaDB :

```
idrisa@ubuntuideidrisa:~$ sudo systemctl stop mariadb
idrisa@ubuntuideidrisa:~$
```

Une fois que c'est fait (et c'est important d'arrêter MariaDB), démarrez l'initialisation du cluster Galera .On va se connecter a notre instance locale MariaDB pour regarder combien de noeuds constituent notre cluster : en toute logique un seul.

```
root@ubuntuideidrisa:/# mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.36-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show variables like 'default_storage_engine';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| default_storage_engine | InnoDB |
+-----+-----+
1 row in set (0,026 sec)

MySQL [(none)]>
```

Exécutez la requête suivante pour récupérer la valeur de la propriété "wsrep\_cluster\_size" :

```
MariaDB [(none)]> show status like 'wsrep_cluster_size';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| wsrep_cluster_size | 1 |
+-----+-----+
1 row in set (0.001 sec)
```

On voit bien que c'est égal a "1", ce qui est une bonne nouvelle .

## V. Ajouter des noeuds au cluster Galère

Premièrement, vous devez configurer le fichier "**60-galera.cnf**" sur les deux autres serveurs, a savoir la machine ***Macdeidy\_SRV-DEB-2*** et ***AmadouDell\_ SRV-DEB-3*** dans mon cas. Copiez-collez la configuration a l'identique, en reprenant le contenu du fichier du premier noeud (modifiez seulement les options "**wsrep\_node\_name**" et "**wsrep\_node\_address**" si vous les utilisez).

,

```
idrissa@MacBook-Pro-de-Idrissa / % mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 21
Server version: 8.2.0 Homebrew
```

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

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> show databases;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| performance_schema      |
| sys                     |
+-----+
4 rows in set (0.04 sec)

mysql> █
```

Dès que le fichier de configuration "**60-galera.cnf**" est prêt , il suffit de redémarrer l'instance MariaDB de **Macdeidy SRV-DEB-2**

pour l'intégrer au cluster :

```
idrissa@MacBook-Pro-de-Idrissa / % brew services restart mariadb

Stopping `mariadb`... (might take a while)
==> Successfully stopped `mariadb` (label: homebrew.mxcl.mariadb)
==> Successfully started `mariadb` (label: homebrew.mxcl.mariadb)
idrissa@MacBook-Pro-de-Idrissa / % █
```

Ensuite, sur le serveur **UBUNTU 22.04\_SRV-DEB-1**, si je regarde le nombre de noeuds présent dans le cluster, j'ai bien la valeur "2". De



plus, si l'on consulte le fichier de log, on peut voir clairement qu'il s'est passé quelque chose et qu'un nouveau noeud a intégré le cluster :

```
MariaDB [(none)]> show status like 'wsrep_cluster_size';
+-----+
| Variable_name | Value |
+-----+
| wsrep_cluster_size | 2 |
+-----+
1 row in set (0.001 sec)

MariaDB [(none)]> Ctrl-C -- exit!
Aborted
root@SRV-DEB-1:~# tail -f /var/log/mysql/error-galera.log
2022-05-09 12:11:55 0 [Note] WSREP: Processing event queue:...100.0% (1/1 events) complete.
2022-05-09 12:11:55 0 [Note] WSREP: Shifting JOINED -> SYNCED (TO: 2)
2022-05-09 12:11:55 1 [Note] WSREP: Server SRV-DEB-1 synced with group
2022-05-09 12:11:55 1 [Note] WSREP: Server status change joined -> synced
2022-05-09 12:11:55 1 [Note] WSREP: Synchronized with group, ready for connections
2022-05-09 12:11:55 1 [Note] WSREP: wsrep_notify_cmd is not defined, skipping notification.
2022-05-09 12:11:57 0 [Note] WSREP: async IST sender served
2022-05-09 12:11:57 0 [Note] WSREP: 0.0 (SRV-DEB-2): State transfer from 1.0 (SRV-DEB-1) complete.
2022-05-09 12:11:57 0 [Note] WSREP: Member 0.0 (SRV-DEB-2) synced with group.
2022-05-09 12:11:57 0 [Note] WSREP: (c917b1d8-93ac, 'tcp://0.0.0.0:4567') turning message relay requesting off
```

Faire la même chose avec la machine *Amadou***DELL\_SRV-DEB-3**

```
root@SRV-DEB-1:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 36
Server version: 10.5.15-MariaDB-0+deb11u1 Debian 11

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show status like 'wsrep_cluster_size';
+-----+
| Variable_name | Value |
+-----+
| wsrep_cluster_size | 3 |
+-----+
1 row in set (0.001 sec)

MariaDB [(none)]> Ctrl-C -- exit!
Aborted
root@SRV-DEB-1:~# tail -f /var/log/mysql/error-galera.log
2022-05-09 12:12:56 1 [Note] WSREP: wsrep_notify_cmd is not defined, skipping notification.
2022-05-09 12:12:56 1 [Note] WSREP: Lowest cert index boundary for CC from group: 3
2022-05-09 12:12:56 1 [Note] WSREP: Min available from gcache for CC from group: 1
2022-05-09 12:12:56 0 [Note] WSREP: Member 1.0 (SRV-DEB-3) requested state transfer from ''any''. Selected 0.0 (SRV-DEB-2) (SYNCED) as donor.
2022-05-09 12:12:57 0 [Note] WSREP: 0.0 (SRV-DEB-2): State transfer to 1.0 (SRV-DEB-3) complete.
2022-05-09 12:12:57 0 [Note] WSREP: Member 0.0 (SRV-DEB-2) synced with group.
2022-05-09 12:12:58 0 [Note] WSREP: (c917b1d8-93ac, 'tcp://0.0.0.0:4567') turning message relay requesting off
2022-05-09 12:12:59 0 [Note] WSREP: 1.0 (SRV-DEB-3): State transfer from 0.0 (SRV-DEB-2) complete.
2022-05-09 12:12:59 0 [Note] WSREP: Member 1.0 (SRV-DEB-3) synced with group.
```

Sur les deux noeuds venant d'être intégrés au cluster, si on liste les bases de données (*show databases;*), nous verrons une nouvelle base de données correspondante a celle répliquée a partir de **UBUNTU**

22.04\_SRV-DEB-1. Désormais, si une base de données est ajoutée sur l'un des nœuds, elle sera synchronisée avec les autres nœuds !

## VI. L'état des noeuds du cluster Galera

Sur chaque noeud, il est possible d'obtenir des informations sur l'état local, notamment en se connectant à l'instance MariaDB et en regardant certaines propriétés du provider "wsrep". La requête ci-dessous donne l'état du noeud, qui normalement doit être **"Synced"**.

```
MariaDB [(none)]> show status like 'wsrep_local_state_comment';
+-----+-----+
Variable_name | Value |
+-----+-----+
wsrep_local_state_comment | Synced |
+-----+-----+
1 row in set (0.001 sec)
```

*Dans le cas où il n'est plus dans la grappe primaire, il ne sera plus sollicité même s'il est en ligne, dans ce cas, il faut simplement relancer le service MariaDB sur ce noeud. Ce phénomène peut se produire si le noeud est isolé à cause d'un problème réseau et qu'il ne parvient plus à contacter les deux autres noeuds de notre cluster à trois noeuds.*

## VII. Comment utiliser le cluster Galera ?

*Nous venons de voir comment mettre en place le cluster Galera afin d'assurer la haute disponibilité de notre base de données, en l'occurrence ici pour une plateforme de Gestion de l'effectivité des enseignements .*

Pour rappel, c'est dans le fichier wp-config.php, que le serveur de base de données se déclare de cette façon :

```
/** MySQL hostname */  
define( 'DB_HOST', 'db.it-connect.tech:3306' );
```

Si l'on met l'adresse IP "192.168.64.2" correspondante a notre noeud SRV-DEB-1, cela signifie qu'en cas de panne du noeud, les autres noeuds seront actifs, mais non utilisés par notre serveur Web, donc on peut dire que le cluster ne sera pas réellement utile. **L'idéal serait d'utiliser une adresse IP virtuelle (VIP) afin que le cluster soit identifiable par une seule adresse IP grâce à un mécanisme d'IP failover.** Pour cela, il existe plusieurs solutions, notamment :

- Keepalived
- MariaDB MaxScale
- HAProxy (en frontal)

Pour finir, sachez que lorsque le serveur MySQL / MariaDB n'est pas situé sur le même serveur que le serveur Web en lui-même, **il faut autoriser les connexions distantes dans MySQL sur chaque nœud.** Pour cela, il faut éditer le fichier de configuration suivant :

```
#bind-address = 127.0.0.1  
bind-address = 0.0.0.0
```

Ensuite, il faut donner des autorisations a l'utilisateur "idrissa" pour administrer la base de données . Ce qui nécessite de se connecter a l'instance MySQL pour créer une autorisation comme ceci

```
[mysql> GRANT ALL privileges ON `GEE`.* TO 'idrissa'@'<192.168.64.2>' IDENTIFIED BY 'baba' WITH GRANT OPTION; FLUSH PRIVILEGES;  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'IDENTIFIED BY 'baba' WITH GRANT OPTION' at line 1  
Query OK, 0 rows affected (0.05 sec)
```

```
mysql>
```

A partir de la, les modifications dans la base de données effectuées sont bien répliquées entre les différents noeuds du cluster.