

La réplication des données avec MySQL

Présentation générale de la réplication

La réplication est supportée par MySQL depuis la version 3.23.15. Elle est unidirectionnelle, c'est-à-dire que les modifications effectuées sur un serveur peuvent être répliquées sur un autre.

- Le serveur source est appelé maître
- et le ou les répliqués sont appelés esclaves.

Un esclave ne peut avoir qu'un maître mais peut aussi être maître d'autres serveurs. Le principe de fonctionnement est le suivant :

Les serveurs MySQL esclaves se connectent au serveur MySQL maître et sont à l'écoute de chaque modification pour que les modifications intervenant sur le serveur MySQL maître puissent être transmises par le serveur MySQL maître, il est nécessaire d'activer sur celui-ci les journaux binaires.

La réplication reproduit les modifications effectuées sur le serveur MySQL maître sur l'ensemble de des serveurs MySQL esclaves. Ceci permet d'augmenter les performances en lecture en augmentant le nombre de répliqués.

Pour avoir une réplication dans les deux sens c'est à dire une configuration maître-maître, il faut définir une réplication maître-esclave bidirectionnelle. Le premier maître est donc l'esclave du second et le second est aussi l'esclave du premier.

Préparation de l'environnement

Installez sur un premier serveur une base de données contenant une seule table. Dans ce tutoriel ce serveur est situé à l'adresse **192.168.1.27**. Ce sera le serveur **MySQL Maître**.

Adaptez les adresses IP de ce tutoriel aux **adresses IP réelles** de vos VMs.

Dans ce tutoriel, votre 2eme serveur LAMP avec Debian a l'adresse IP 192.168.1.18 et ne contient aucune base de données GLPI. Ce sera le serveur **MySQL esclave**.

Mise en place d'une réplication Maître / Esclave

Créez un utilisateur spécial pour la réplication sur le maître avec les droits de REPLICATION SLAVE. Vous devez aussi lui donner les droits de connexion depuis tous les esclaves. Comme cet l'utilisateur ne fait que de la réplication (ce qui est recommandé), vous n'avez pas à lui donner d'autres droits. Donnez comme nom **repl**:

Configuration du serveur MySQL Maître

Ouvrez une console sur le serveur MySQL Maître et tapez les commandes suivantes :

```
# mysql -u root -pbtssio
mysql> GRANT REPLICATION SLAVE on *.* to 'repl'@'%' identified by 'btssio';
```

Modifiez le fichier de configuration `/etc/mysql/my.cnf` du serveur MySQL Maître pour ajouter la section `[mysqld]` :

```
bind-address      = 0.0.0.0
server-id         = 1
log-bin           = /var/log/mysql/mysql-bin.log
```

Commentaires :

- En positionnant la variable `bind-address` à `0.0.0.0` vous forcez MySQL à écouter sur toutes les interfaces réseau (ou une des IPs locales si l'on veut que les données MySQL ne transi-tent que sur une interface).
- Il faut par contre absolument activer les journaux binaires sur le serveur maître en rajoutant l'option `log-bin`.
- Renseignez la variable **server-id** de manière unique sur le maître (par exemple en lui donnant la valeur 1) et son **esclave** (par exemple en lui donnant la valeur 10).

Redémarrez le serveur MySQL Maître:

```
# service mysql restart
```

Vérification qu'il y ait bien un log binaire :

```
# mysql -u root -pbtssio
mysql> SHOW MASTER STATUS;
+-----+-----+-----+-----+
| File           | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+-----+-----+-----+-----+
| mysql-bin.000001 | 28275   |              |                  |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

Il y a bien un log binaire appelé ici **mysql-bin.000001**.

Configuration du serveur MySQL Esclave

Configurer le serveur MySQL Esclave pour qu'il écoute sur le réseau :

Modifiez le fichier de configuration **/etc/mysql/my.cnf** du serveur MySQL Esclave dans la section **[mysqld]** :

```
bind-address      = 0.0.0.0
server-id         = 10
```

Commentaires

- Rappel il faut positionner la variable **server-id** de manière unique sur le maître et son esclave.

Effectuer une sauvegarde sur le maître

Toute sauvegarde effectuée sur le maître doit s'accompagner d'un relevé de la position dans les journaux binaires. En effet, les esclaves qui seront configurés par la suite ont besoin de savoir où commencer à récupérer les ordres ou lignes à répliquer. Dans le cas d'une sauvegarde logique basée sur mysqldump on utilisera l'option **-master-data** qui relève les informations de position (**MASTERLOGFILE** et **MASTERLOGPOS**) et ajoute la commande **CHANGE MASTER TO** adéquate dans le dump SQL.

```
# mysqldump -u root -p --master-data --single-transaction --all-databases > dump.sql
```

Le mot de passe du compte root de MySQL vous sera demandé.

Restaurer la sauvegarde sur l'esclave :

Il ne reste plus qu'à restaurer ces données sur le serveur MySQL Esclave (et tous les autres esclaves éventuels) :

- Transférer avec winscp le fichier dump.sql sur le serveur esclave
- Restaurer les bases avec la commande suivante ; **Attention** , cela va écraser toutes les base de données présentes :

```
# mysql -u root -pbtssio < dump.sql
```

Finaliser la configuration de l'esclave :

Il reste à positionner les variables **MASTERHOST**, **MASTERUSER** et **MASTER_PASSWORD** :

Sur le serveur MySQL Esclave, ouvrez une session console et tapez les commandes suivantes :

```
# mysql -u root -pbtssio
mysql> CHANGE MASTER TO MASTER_HOST='192.168.1.27',
MASTER_USER='repl',
MASTER_PASSWORD='btssio';
```

Redémarrez le serveur MySQL esclave :

```
# /etc/init.d/mysql restart
```

Si nécessaire lancez le serveur MySQL Esclave :

```
# mysql -u root -pbtssio
mysql> START SLAVE;
```

Avec Phpmyadmin, vous pouvez voir qu'une colonne **Réplication** qui montre que la réplication est active pour toutes les tables :

Pour vérifier que la synchronisation fonctionne : Normalement, s'il n'y a pas de problème de configuration, les 2 serveurs ont redémarré sans erreurs. Pour vérifier que la réplication s'effectue correctement, lancez la commande suivante sur l'esclave :

```
# mysql -u root -pbtssio
mysql> SHOW SLAVE STATUS;
```

Il y a 2 informations à vérifier :

Secondsbehindmaster : Celle-ci doit être à 0. Si elle indique Null, dans ce cas, la réplication ne fonctionne pas. Vérifiez alors l'information Last/IOError. Cette vérification n'est pas facile à faire. Par contre vous pouvez vous assurer qu'il n'y a pas de message d'erreur renvoyé.

L'une des causes fréquentes d'erreur au début est un problème de mot de passe pour l'utilisateur de réplication.

À présent, les modifications effectuées sur le serveur MySQL Maître se retrouveront répliquées sur le serveur MySQL Esclave.

Pour le vérifier :

- Avec PhpMyadmin, créez un nouvel enregistrement dans la table de la base de données sur le serveur MySQL Maître (adresse IP 192.168.1.27).
- Avec Phpmyadmin, vous devez visualiser maintenant dans la table du serveur MySQL esclave (adresse IP 192.168.1.18) que l'enregistrement a été répliqué.

Vous pouvez aussi directement lancer une commande **INSERT, UPDATE ou DELETE** sur le serveur maître et vérifiez que les données apparaissent bien sur l'esclave.

En cas de problème

Si la réplication **ne fonctionne pas** ou **ne fonctionne plus**, vous pouvez essayer la démarche suivante :

- Refaites une sauvegarde depuis le serveur Maître

```
# mysqldump -u root -pbtssio --master-data --single-transaction --all-databases > dump.sql
```

- arrêtez la réplication sur le serveur Esclave

```
# mysql -u root -pbtssio
mysql> STOP SLAVE;
quit
```

- Restaurez la sauvegarde sur le serveur Esclave

```
# mysql -u root -pbtssio < dump.sql
```

- Relancez le serveur Esclave

```
# mysql -u root -pbtssio
mysql> START SLAVE;
```

Retour au dossier sur la réplication des données ...

[Fiche Cours \(C6\) : La réplication des données](#)

From:

/ - **Les cours du BTS SIO**

Permanent link:

</doku.php/si7/a10>

Last update: **2018/03/08 15:46**

