

# La Réplication

Pour faire simple, faire la réplication entre deux VM revient à les associer. Quand on passe une commande sur le serveur principal, ça la passe aussi sur le serveur secondaire (esclave).

Pour faire de la réplication entre deux serveurs, on commence tous d'abord par ajouté les modifications suivantes sur les serveurs (Dans mon cas des databases) :

Sur les deux databases

Tout d'abord, il faut créer un dossier que l'on nommera "mysql" dans le chemin /var/log de la database

```
btssio@Debian-Datal:/var/log$ sudo mkdir mysql
[sudo] Mot de passe de btssio :
btssio@Debian-Datal:/var/log$ ls
alternatives.log  apache2  btmp      dpkg.log  faillog   journal  mysql  README  wtmp
alternatives.log.1 apt       btmp.1    dpkg.log.1 installer lastlog  private runit
```

Après avoir créé le dossier "mysql" il faut lui attribuer des droits (mettre chmod 770) :

```
btssio@Debian-Datal:/var/log$ sudo chmod 777 mysql/
btssio@Debian-Datal:/var/log$ ls -al
total 372
drwxr-xr-x  9 root root          4096  6 nov.  13:24 .
drwxr-xr-x 12 root root          4096 18 oct.  12:28 ..
-rw-r--r--  1 root root           0  5 nov.  14:37 alternatives.log
-rw-r--r--  1 root root        14406 18 oct.  12:29 alternatives.log.1
drwxr-x---  2 root adm           4096  8 nov.  10:25 apache2
drwxr-xr-x  2 root root           4096  5 nov.  14:37 apt
-rw-rw----  1 root utmp           768  5 nov.  14:37 btmp
-rw-rw----  1 root utmp           384 15 oct.  09:24 btmp.1
-rw-r--r--  1 root root           0  5 nov.  14:37 dpkg.log
-rw-r--r--  1 root root       252646 18 oct.  12:30 dpkg.log.1
-rw-r--r--  1 root root           0 11 sept. 09:19 faillog
drwxr-xr-x  3 root root           4096 11 sept. 09:27 installer
drwxr-sr-x+ 3 root systemd-journal 4096 11 sept. 09:28 journal
-rw-rw-r--  1 root utmp       292292  8 nov.  10:25 lastlog
drwxrwxrwx  2 root root           4096  8 nov.  10:40 mysql
```

Sur le serveur principal

```
btssio@Debian-Data1:/etc/mysql/mariadb.conf.d$ sudo nano 50-server.cnf
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
# other settings you may need to change.
#server-id             = 1
#log_bin               = /var/log/mysql/mysql-bin.log
expire_logs_days       = 10
#max_binlog_size       = 100M

log_bin = /var/log/mysql/mariadb-bin
log_bin_index = /var/log/mysql/mariadb-bin.index
serv-id = 1

#
# * SSL/TLS
#
```

### Sur le serveur secondaire

```
btssio@Debian-Data2:/etc/mysql/mariadb.conf.d$ sudo nano 50-server.cnf
# The following can be used as easy to replay backup logs or for replication.
# note: if you are setting up a replication slave, see README.Debian about
# other settings you may need to change.
#server-id             = 1
#log_bin               = /var/log/mysql/mysql-bin.log
expire_logs_days       = 10
#max_binlog_size       = 100M

relay_log = /var/log/mysql/mariadb-relay-bin
relay_log_index = /var/log/mysql/mariadb-relay-bin.index
server-id = 2

#
# * SSL/TLS
#
```

### Sur le serveur principal

Une fois cette étape accomplie, on doit créer un utilisateur de réplication sur le serveur principal. Pour ce faire on utilise la commande :

#mysql -u (nom de l'utilisateur) -p (le -p est optionnel si on est en root)

```
root@Debian-Data1:~# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 32
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12

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)]>
```

Maintenant qu'on est dans mariadb, on peut créer l'utilisateur de réplication comme ci-dessous :

```
MariaDB [(none)]> create user 'replication'@'172.21.3.13' identified by 'sisr2'
-> ;
Query OK, 0 rows affected (0,003 sec)

MariaDB [(none)]> grant replication slave on *.* to 'replication'@'172.21.3.13';
Query OK, 0 rows affected (0,001 sec)
```

### Sur le serveur secondaire

Dans mariadb taper la commande suivante :

```
MariaDB [(none)]> CHANGE MASTER TO MASTER_HOST='172.21.3.12', MASTER_USER='replication', MASTER_PASSWO
RD='sisr2', MASTER_LOG_FILE='mariadb-bin.000001', MASTER_LOG_POS=678;
Query OK, 0 rows affected, 1 warning (0,025 sec)
```

Pour savoir quoi mettre dans “MASTER\_LOG\_FILE” et “MASTER\_LOG\_POS” il faut se rendre sur le serveur principal aller dans mariadb et taper la commande suivante :

```
MariaDB [(none)]> show master status
-> ;
```

File	Position	Binlog_Do_DB	Binlog_Ignore_DB
mariadb-bin.000001	678		

```
1 row in set (0,000 sec)
```

Une fois ces étapes réaliser un peu lancer le serveur secondaire comme ci-dessous :

```
MariaDB [(none)]> start slave;
Query OK, 0 rows affected (0,002 sec)
```

### Général

Pour que la réplication fonctionne entre deux serveurs, il faut que les deux serveurs aient les mêmes informations de base. Le serveur secondaire ne peut pas copier des éléments si les données ne sont pas identiques à la base.

Pour faire en sorte que les deux serveurs aient les mêmes informations, il faut soit faire la réplication avant de toucher a mariadb ou alors nous pouvons utiliser MySQL dump coupler a win-scp pour copier les informations de bases d'un serveur a l'autre.'

NB : Pour pouvoir faire le dump d'un fichier il faut qu'un utilisateur localhost est les droits sur le fichier / la base de données que l'on veut dump.

```

MariaDB [(none)]> create user 'dolibarr'@'localhost' identified by 'sisr2';
Query OK, 0 rows affected (0,003 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> grant all privileges on dolibarr_db.* to 'dolibarr'@'localhost';
Query OK, 0 rows affected (0,003 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0,001 sec)

```

```
btssio@Debian-Datal:~$ mysqldump -u dolibarr -psisr2 dolibarr_db > /home/btssio/dolibarr.sql
```

```

MariaDB [(none)]> create user 'atedilocal'@'localhost' identified by 'sisr2';
Query OK, 0 rows affected (0,002 sec)

MariaDB [(none)]> grant all privileges on atedi_db.* to 'atedilocal'@'localhost';
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0,001 sec)

```

```
btssio@Debian-Datal:~$ mysqldump -u atedilocal -psisr2 atedi_db > /home/btssio/atedi.sql
```

```

btssio@Debian-Datal:~$ ls
atedi_backup.sql  atedi.sql  backup.sql  dolibarr.sql  script

```

Après avoir fait le dump des différents fichiers en se rend sur Win-SCP, on se connecte sur le serveur principal puis sur le secondaire, on cherche les dossiers du serveur principal qu'on a 'dump' (Ici atedi.sql et dolibarr.sql) et on les copie sur le serveur secondaire.

