

RAPPORT SUR LA PARTIE BASE DE DONNEES MYSQL REPARTIES SUR 3 NŒUDS EN UTILISANT PROXYSQL :

I. Installation ProxySQL :

On a installé ProxySQL via le site <https://proxysql.com/> , après installation de ProxySQL on installe le package avec **dpkg** qui est utilisé pour gérer les packages logiciels .deb le **-i** indique que nous aimerions installer à partir du fichier spécifié **sudo dpkg -i nom_du_fichier**
Ensuite, nous aurons besoin d'une application cliente MySQL pour se connecter à l'instance ProxySQL. En effet, ProxySQL utilise en interne une interface compatible avec MySQL pour les tâches administratives. Nous utiliserons l'outil mysql de ligne de commande, qui fait partie du **mysql-clientpackage** disponible dans les référentiels Ubuntu.
Mettez à jour votre référentiel de packages pour vous assurer que vous disposez de la dernière version pré-intégrée, puis installez le **mysql-clientpackage**.

```
sudo apt-get update
```

```
sudo apt-get install mysql-client
```

Vous avez maintenant toutes les conditions requises pour exécuter ProxySQL, mais le service ne démarre pas automatiquement après l'installation, alors démarrez-le manuellement maintenant.

```
sudo systemctl start proxysql
```

ProxySQL devrait maintenant fonctionner avec sa configuration par défaut en place. Vous pouvez vérifier en utilisant **systemctl**.

```
systemctl status proxysql
```

La sortie ressemblera à ceci :

```

abdoulaye@abdoulaye-VirtualBox:~/Téléchargements$ sudo systemctl start proxysql
abdoulaye@abdoulaye-VirtualBox:~/Téléchargements$ systemctl status proxysql
● proxysql.service - High Performance Advanced Proxy for MySQL
   Loaded: loaded (/lib/systemd/system/proxysql.service; enabled; vendor pre>
   Active: active (running) since Fri 2023-03-17 01:52:24 GMT; 29s ago
   Process: 5087 ExecStart=/usr/bin/proxysql --idle-threads -c /etc/proxysql.>
  Main PID: 5089 (proxysql)
     Tasks: 25 (limit: 2043)
    Memory: 11.8M
    CGroup: /system.slice/proxysql.service
            └─5089 /usr/bin/proxysql --idle-threads -c /etc/proxysql.cnf
              └─5090 /usr/bin/proxysql --idle-threads -c /etc/proxysql.cnf

mar 17 01:52:23 abdoulaye-VirtualBox systemd[1]: Starting High Performance Adv>
mar 17 01:52:23 abdoulaye-VirtualBox proxysql[5087]: 2023-03-17 01:52:23 [INFO>
mar 17 01:52:23 abdoulaye-VirtualBox proxysql[5087]: 2023-03-17 01:52:23 [INFO>
mar 17 01:52:23 abdoulaye-VirtualBox proxysql[5087]: 2023-03-17 01:52:23 [INFO>
mar 17 01:52:23 abdoulaye-VirtualBox proxysql[5087]: 2023-03-17 01:52:23 [INFO>
mar 17 01:52:24 abdoulaye-VirtualBox systemd[1]: Started High Performance Adva>
lines 1-17/17 (END)

```

La active (running) ligne signifie que ProxySQL est installé et en cours d'exécution.

Ensuite, nous augmenterons la sécurité en définissant le mot de passe utilisé pour accéder à l'interface d'administration de ProxySQL.

II. Définition du mot de passe de l'administrateur ProxySQL :

La première fois que vous démarrez une nouvelle installation de ProxySQL, il utilise un fichier de configuration fourni par le package pour initialiser les valeurs par défaut de toutes ses variables de configuration. Après cette initialisation, ProxySQL stocke sa configuration dans une base de données que vous pouvez gérer et modifier via la ligne de commande.

Pour définir le mot de passe administrateur dans ProxySQL, nous allons nous connecter à cette base de données de configuration et mettre à jour les variables appropriées.

Tout d'abord, accédez à l'interface d'administration. Vous serez invité à entrer le mot de passe qui, sur une installation par défaut, est **admin**.

```
mysql -u admin -p -h 127.0.0.1 -P 6032 --prompt='ProxySQLAdmin> '
```

- **-u** spécifie l'utilisateur avec lequel nous voulons nous connecter, ici **admin**, l'utilisateur par défaut pour les tâches administratives telles que la modification des paramètres de configuration.
- **-h 127.0.0.1** indique mysql de se connecter à l'instance ProxySQL locale. Nous devons définir cela explicitement car ProxySQL n'écoute pas le fichier socket que mysql suppose par défaut.

- **-P** spécifie le port auquel se connecter. L'interface d'administration de ProxySQL écoute sur **6032**.
- **--prompt** est un indicateur facultatif qui modifie l'invite par défaut, qui est normalement **mysql>**. Ici, nous le remplaçons par **ProxySQLAdmin>** pour indiquer clairement que nous sommes connectés à l'interface d'administration de ProxySQL. Cela sera utile pour éviter toute confusion plus tard lorsque nous nous connecterons également aux interfaces MySQL sur les serveurs de bases de données répliquées...

Une fois connecté, vous verrez l'invite **ProxySQLAdmin>** :

```
abdoulaye@abdoulaye-VirtualBox:~$ mysql -u admin -p -h 127.0.0.1 -P 6032 --prompt='ProxySQLAdmin>'
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.5.30 (ProxySQL Admin Module)

Copyright (c) 2000, 2023, 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.
ProxySQLAdmin>
```

Modifiez le mot de passe du compte administratif en mettant à jour (UPDATE) la **admin-admin_credentials** variable de configuration dans la **global_variables** base de données.

```
ProxySQLAdmin>UPDATE global_variables SET variable_value='admin:password' WHERE
variable_name='admin-admin_credentials';
Query OK, 1 row affected (0,01 sec)
```

Cette modification ne prendra pas effet immédiatement en raison du fonctionnement du système de configuration de ProxySQL. Il se compose de trois couches distinctes :

- **memory**, qui est modifié lors des modifications à partir de l'interface de ligne de commande.
- **runtime**, qui est utilisé par ProxySQL comme configuration effective.
- **disk**, qui est utilisé pour rendre une configuration persistante après les redémarrages.

À l'heure actuelle, la modification que vous avez apportée est en **mémoire**. Pour appliquer la modification, vous devez copier les paramètres **de mémoire** dans le domaine **d'exécution**, puis les enregistrer sur **le disque** pour les rendre persistants.

LOAD ADMIN VARIABLES TO RUNTIME;

SAVE ADMIN VARIABLES TO DISK;

Ces **ADMIN** commandes gèrent uniquement les variables liées à l'interface de ligne de commande d'administration. ProxySQL expose des commandes similaires, comme MySQL, pour gérer d'autres parties de sa configuration. Nous les utiliserons plus tard dans ce tutoriel. Maintenant que ProxySQL est installé et fonctionne avec un nouveau mot de passe administrateur, configurons les 3 nœuds MySQL afin que ProxySQL puisse les surveiller. Gardez l'interface ProxySQL ouverte, car nous l'utiliserons plus tard.

III. Configuration de la surveillance dans MySQL :

ProxySQL doit communiquer avec les nœuds MySQL pour pouvoir évaluer leur état. Pour cela, il doit pouvoir se connecter à chaque serveur avec un utilisateur dédié.

Ici, nous allons configurer l'utilisateur nécessaire sur les nœuds MySQL et installer des fonctions SQL supplémentaires qui permettent à ProxySQL d'interroger l'état de réplication du groupe. Étant donné que la réplication de groupe MySQL est déjà en cours d'exécution, les étapes suivantes ne doivent être effectuées que sur **un seul membre du groupe**.

Dans un deuxième terminal, téléchargez le fichier SQL contenant certaines fonctions nécessaires pour que la prise en charge de la réplication de groupe ProxySQL fonctionne.

curl -OL

https://gist.github.com/lefred/77ddbde301c72535381ae7af9f968322/raw/5e40b03333a3c148b78aa348fd2cd5b5dbb36e4d/addition_to_sys.sql

```
abdoulaye@abdoulaye-VirtualBox:~$ curl -OL https://gist.github.com/lefred/77ddbde301c72535381ae7af9f968322/raw/5e40b03333a3c148b78aa348fd2cd5b5dbb36e4d/addition_to_sys.sql
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           %             %             Dload  Upload  Total   Spent    Left     Speed
  0     0    0     0    0     0      0      0  --:--:--  0:00:04  --:--:--    0
100 2647 100 2647    0     0    194      0  0:00:13  0:00:13  --:--:--   578
abdoulaye@abdoulaye-VirtualBox:~$ less addition_to_sys.sql
abdoulaye@abdoulaye-VirtualBox:~$ sudo mysql -u root -p < addition_to_sys.sql
[sudo] Mot de passe de abdoulaye :
Enter password:
abdoulaye@abdoulaye-VirtualBox:~$ mysql -u root -p
```

Vous pouvez afficher le contenu du fichier à l'aide de **less addition_to_sys.sql**.

Lorsque vous êtes prêt, exécutez les commandes du fichier. Vous serez invité à saisir le mot de passe d'administration MySQL.

mysql -u root -p < addition_to_sys.sql

Si la commande s'exécute avec succès, elle ne produira aucune sortie. Dans ce cas, tous les nœuds MySQL exposeront désormais les fonctions nécessaires pour que ProxySQL reconnaisse l'état de réplication du groupe.

Ensuite, nous devons créer un utilisateur dédié qui sera utilisé par ProxySQL pour surveiller la santé des instances.

Ouvrez l'invite interactive MySQL, qui vous demandera à nouveau le mot de passe **root**.

mysql -u root -p

Créez ensuite l'utilisateur dédié, que nous appelons **moniteur** ici.

```
CREATE USER 'monitor'@'%' IDENTIFIED BY 'monitorpassword';
```

Accordez les privilèges d'utilisateur pour interroger l'état du serveur MySQL à l'utilisateur **du moniteur** .

```
GRANT SELECT on sys.* to 'monitor'@'%';
```

Enfin, appliquez les modifications.

```
FLUSH PRIVILEGES;
```

En raison de la réplication de groupe, une fois que vous avez terminé d'ajouter l'utilisateur pour la surveillance de la santé à un nœud MySQL, il sera entièrement configuré sur les trois nœuds. Ensuite, nous devons mettre à jour ProxySQL avec les informations de cet utilisateur afin qu'il puisse accéder aux nœuds MySQL.

IV. Configuration de la surveillance dans ProxySQL :

Pour configurer ProxySQL pour utiliser le nouveau compte d'utilisateur lors de la surveillance des nœuds, nous allons utiliser UPDATE la variable de configuration appropriée. Ceci est très similaire à la façon dont nous définissons le mot de passe administrateur de l'étape 2.

De retour dans l'interface d'administration de ProxySQL, mettez à jour la mysql-monitor_usernamevariable avec le nom d'utilisateur du nouveau compte.

```
ProsySQLAdmin>UPDATE global_variables SET variable_value='monitor' WHERE variable_name='mysql-monitor_username';
Query OK, 1 row affected (0,27 sec)

ProsySQLAdmin>LOAD MYSQL VARIABLES TO RUNTIME;
Query OK, 0 rows affected (0,12 sec)

ProsySQLAdmin>SAVE MYSQL VARIABLES TO DISK;
Query OK, 155 rows affected (0,11 sec)

ProsySQLAdmin>
```

Comme auparavant, la configuration n'est pas automatiquement appliquée, alors migrez-la dans **l'environnement d'exécution** et enregistrez-la sur **le disque** . Cette fois, notez que nous utilisons MYSQL au lieu de ADMIN pour mettre à jour ces variables car nous modifions les variables de configuration MySQL.

```
LOAD MYSQL VARIABLES TO RUNTIME;
```

```
SAVE MYSQL VARIABLES TO DISK;
```

Le compte de surveillance est configuré à toutes les extrémités, et l'étape suivante consiste à informer ProxySQL des nœuds eux-mêmes.

V. Ajout de nœuds MySQL au pool de serveurs ProxySQL :

Pour que ProxySQL connaisse nos trois nœuds MySQL, nous devons indiquer à ProxySQL comment les répartir sur ses groupes d'hôtes, qui sont des ensembles de nœuds désignés. Chaque groupe d'hôtes est identifié par un nombre positif, comme 1ou 2. Les groupes d'hôtes peuvent acheminer différentes requêtes SQL vers différents ensembles d'hôtes lors de l'utilisation du routage de requête ProxySQL.

Dans les configurations de réplication statique, les groupes d'hôtes peuvent être définis arbitrairement. Cependant, la prise en charge de la réplication de groupe de ProxySQL divise automatiquement tous les nœuds d'un groupe de réplication en quatre états logiques :

writers, qui sont des nœuds MySQL pouvant accepter des requêtes qui modifient les données. ProxySQL s'assure de maintenir tous les nœuds principaux jusqu'à la quantité maximale définie dans ce groupe.

Les rédacteurs de sauvegarde, qui sont également des nœuds MySQL qui peuvent accepter des requêtes qui modifient les données. Cependant, ces nœuds ne sont pas désignés comme écrivains ; les nœuds principaux dépassant le nombre défini d'enregistreurs maintenus sont conservés dans ce groupe et sont promus en enregistreurs si l'un des enregistreurs échoue.

Les lecteurs sont des nœuds MySQL qui ne peuvent pas accepter les requêtes modifiant les données et doivent être utilisés comme nœuds en lecture seule. ProxySQL ne place ici que des nœuds esclaves.

Offline, qui concerne les nœuds qui se comportent mal en raison de problèmes tels qu'un manque de connectivité ou un trafic lent.

Chacun de ces quatre états a des groupes d'hôtes correspondants, mais les identificateurs de groupe numériques ne sont pas attribués automatiquement.

En mettant tout cela ensemble, nous devons dire à ProxySQL quels identifiants il doit utiliser pour chaque état. Ici, nous utilisons 1 pour le groupe d'hôtes **hors ligne**, 2 pour le groupe d'hôtes d'écriture, 3 pour le groupe d'hôtes de lecture et 4 pour le groupe d'hôtes d'écriture de sauvegarde.

Pour définir ces identifiants, créez une nouvelle ligne avec ces variables et valeurs dans la **mysql_group_replication_hostgroup** table de configuration.

Voici les variables supplémentaires définies dans cette ligne et ce que chacune fait :

- **active** défini pour 1 activer la surveillance par ProxySQL de ces groupes d'hôtes.
- **max_writers** définit combien de nœuds peuvent agir en tant qu'écrivains. Nous avons utilisé 3 ici parce que dans une configuration multi-primaire, tous les nœuds peuvent être traités de la même manière, nous avons donc utilisé ici 3 (le nombre total de nœuds).
- **writer_is_also_readers** et to 1 demande à ProxySQL de traiter également les rédacteurs comme des lecteurs.
- **max_transactions_behind** définit le nombre maximum de transactions retardées avant qu'un nœud ne soit classé comme étant **hors ligne**.

Maintenant que ProxySQL sait comment répartir les nœuds entre les groupes d'hôtes, nous pouvons ajouter nos serveurs MySQL au pool. Pour ce faire, nous avons besoin de INSERT l'adresse IP et du groupe d'hôtes initial de chaque serveur dans la **mysql_server** table, qui contient la liste des serveurs avec lesquels ProxySQL peut interagir.

Ajoutez chacun des trois serveurs MySQL, en veillant à remplacer les exemples d'adresses IP dans les commandes ci-dessous.

```

ProsySQLAdmin>INSERT INTO mysql_group_replication_hostgroups (writer_hostgroup,
  backup_writer_hostgroup, reader_hostgroup, offline_hostgroup, active, max_writ
  ers, writer_is_also_reader, max_transactions_behind) VALUES (2, 4, 3, 1, 1, 3,
  1, 100);
Query OK, 1 row affected (0,13 sec)

ProsySQLAdmin>INSERT INTO mysql_servers(hostgroup_id, hostname, port) VALUES (2
  , '203.0.113.1', 3306);
Query OK, 1 row affected (0,12 sec)

ProsySQLAdmin>INSERT INTO mysql_servers(hostgroup_id, hostname, port) VALUES (2
  , '203.0.113.2', 3306);
Query OK, 1 row affected (0,01 sec)

ProsySQLAdmin>INSERT INTO mysql_servers(hostgroup_id, hostname, port) VALUES (2
  , '203.0.113.3', 3306);
Query OK, 1 row affected (0,00 sec)

ProsySQLAdmin>

```

Ici, la 2valeur définit initialement tous ces nœuds comme écrivains et 3306définit le port MySQL par défaut.

Comme auparavant, migrez ces modifications dans **l'environnement d'exécution** et enregistrez-les sur **le disque** pour appliquer les modifications.

ProxySQL devrait maintenant distribuer nos nœuds sur les groupes d'hôtes comme spécifié. Vérifions cela en exécutant une SELECTrequête sur la runtime_mysql_serverstable, qui expose l'état actuel des serveurs que ProxySQL utilise.

```

ProsySQLAdmin>INSERT INTO mysql_servers(hostgroup_id, hostname, port) VALUES (2
  , '203.0.113.3', 3306);
Query OK, 1 row affected (0,00 sec)

ProsySQLAdmin>LOAD MYSQL SERVERS TO RUNTIME;
Query OK, 0 rows affected (0,57 sec)

ProsySQLAdmin>SAVE MYSQL SERVERS TO DISK;
Query OK, 0 rows affected (0,26 sec)

ProsySQLAdmin>SELECT hostgroup_id, hostname, status FROM runtime_mysql_servers;

+-----+-----+-----+
| hostgroup_id | hostname      | status |
+-----+-----+-----+
| 1            | 203.0.113.1  | ONLINE |
| 1            | 203.0.113.2  | ONLINE |
| 1            | 203.0.113.3  | ONLINE |
+-----+-----+-----+
3 rows in set (0,19 sec)

```

Cependant, avant de pouvoir les utiliser, nous devons configurer les informations d'identification de l'utilisateur pour accéder aux bases de données MySQL sur chaque nœud.

VI. Création des utilisateurs MySQL :

ProxySQL agit comme un équilibreur de charge ; les utilisateurs finaux se connectent à ProxySQL, et ProxySQL transmet à son tour la connexion au nœud MySQL choisi. Pour se

connecter à un nœud individuel, ProxySQL réutilise les informations d'identification avec lesquelles il a été accédé.

Pour autoriser l'accès aux bases de données situées sur les nœuds de réplication, nous devons créer un compte utilisateur avec les mêmes informations d'identification que ProxySQL et accorder à cet utilisateur les privilèges nécessaires.

Comme à l'étape 3, les étapes suivantes ne doivent être effectuées que sur un seul membre du groupe. Vous pouvez choisir n'importe quel membre.

Créez un nouvel utilisateur appelé **playgrounduser** identifié avec le mot de passe **playgroundpassword**.

Donnez-lui des privilèges pour accéder pleinement à la playgroundbase de données de test à partir du didacticiel de réplication de groupe d'origine .

Appliquez ensuite les modifications et quittez l'invite.

```
mysql> CREATE USER 'playgrounduser'@'%' IDENTIFIED BY 'playgroundpassword';
Query OK, 0 rows affected (0,09 sec)

mysql> GRANT ALL PRIVILEGES on playground.* to 'playgrounduser'@'%;
Query OK, 0 rows affected (0,02 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,01 sec)

mysql> EXIT;
Bye
abdoulaye@abdoulaye-VirtualBox:~$
```

Vous pouvez vérifier que l'utilisateur a été correctement créé en essayant d'accéder à la base de données avec les informations d'identification nouvellement configurées directement sur le nœud.

Rouvrez l'interface MySQL avec le nouvel utilisateur, qui vous demandera le mot de passe.

```
abdoulaye@abdoulaye-VirtualBox:~$ mysql -u playgrounduser -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 22
Server version: 8.0.32-0ubuntu0.20.04.2 (Ubuntu)

Copyright (c) 2000, 2023, 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.
```

VII. Création de l'utilisateur ProxySQL :

La dernière étape de configuration consiste à autoriser les connexions à ProxySQL avec l'utilisateur playgrounduser et à transmettre ces connexions aux nœuds.

Pour ce faire, nous devons définir des variables de configuration dans la **mysql_userstable**, qui contient les informations d'identification de l'utilisateur. Dans l'interface ProxySQL, ajoutez le nom d'utilisateur, le mot de passe et le groupe d'hôtes par défaut à la base de données de configuration (qui est 2, pour le groupe d'hôtes de l'écrivain)

Migrez la configuration dans **l'environnement d'exécution** et enregistrez-la sur le **disque** pour appliquer la nouvelle configuration.

```
ProxySQLAdmin>INSERT INTO mysql_users(username, password, default_hostgroup) VALUES ('playgrounduser', 'playgroundpassword', 2);
Query OK, 1 row affected (0,07 sec)

ProxySQLAdmin>LOAD MYSQL USERS TO RUNTIME;
Query OK, 0 rows affected (0,03 sec)

ProxySQLAdmin>SAVE MYSQL USERS TO DISK;
Query OK, 0 rows affected (0,09 sec)

ProxySQLAdmin>
```

ProxySQL écoute sur le port 6033 les connexions client entrantes, essayez donc de vous connecter à la vraie base de données (pas à l'interface d'administration) en utilisant **playgrounduser** et port 6033. Vous serez invité à entrer le mot de passe, qui était **playgroundpassword** dans notre exemple.

Ici, nous définissons l'invite sur ProxySQLClient> afin de pouvoir la distinguer de l'invite de l'interface d'administration. Nous utiliserons les deux lors du test de la configuration finale.

L'invite devrait s'ouvrir, ce qui signifie que les informations d'identification ont été acceptées par ProxySQL lui-même.

```
abdoulaye@abdoulaye-VirtualBox:~$ mysql -u playgrounduser -p -h 127.0.0.1 -P 6033 --prompt='ProxySQLClient> '
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 5.5.30 (ProxySQL)

Copyright (c) 2000, 2023, 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.
ProxySQLClient>
```

Enfin créons la base de données hotellerie avec ses attributs.

```
mysql> use hotellerie;
Database changed
mysql> show tables;
+-----+
| Tables_in_hotellerie |
+-----+
| categorie              |
| chambre                |
| client                 |
| facture                |
| hotel                  |
| reservation            |
| services               |
+-----+
7 rows in set (0,01 sec)

mysql> █
```