Rapport sur la Persistance des Données MySQL via Volumes Docker

Auteur : Ambinintsoa RANDRIANARISOA

2 décembre 2025

Résumé

Ce rapport documente la procédure d'utilisation des volumes nommés de Docker pour assurer la persistance des données d'un service MySQL. L'objectif était de créer une base de données, d'insérer des enregistrements, puis de prouver que ces données restent accessibles même après la suppression du conteneur initial et le lancement d'un nouveau conteneur partageant le même volume.

1 Démarrage du Conteneur et Création du Volume Persistant

Il faut initialiser le service MySQL en utilisant un volume nommé, ce qui est l'étape cruciale pour garantir la persistance des données.

Instruction 1 : Lancement et Création du Volume

Il faut lancer le conteneur `mysql:8` en mode détaché et lui attribuer le volume nommé `mysql_data`.

```
docker run -d \
  --name mysql-original \
  -e MYSQL_ROOT_PASSWORD=pass \
  -v mysql_data:/var/lib/mysql \
  mysql:8
```

Ce volume est monté sur le répertoire `/var/lib/mysql` à l'intérieur du conteneur, où MySQL stocke ses fichiers de base de données.

2 Création de la Base et Insertion des Données

Une fois le conteneur démarré (une vérification des logs est recommandée), il faut se connecter et exécuter les commandes SQL de création et d'insertion.

Instruction 2 : Connexion et Exécution SQL

Il faut exécuter les commandes suivantes directement via `docker exec` pour créer la base `mon_projet`, la table `utilisateurs` et insérer deux enregistrements.

```
docker exec -it mysql-original mysql -u root -ppass -e "
  -- Création de la base de données
  CREATE DATABASE IF NOT EXISTS mon_projet;
```

```
4
5     -- Utilisation de la base de données
6     USE mon_projet;
7
8     -- Création d'une table
9     CREATE TABLE utilisateurs (
10      id INT AUTO_INCREMENT PRIMARY KEY,
11      nom VARCHAR(100),
12      email VARCHAR(100)
13    );
14
15    -- Insertion de données
16    INSERT INTO utilisateurs (nom, email) VALUES
17      ('Alice', 'alice@example.com'),
18      ('Bob', 'bob@example.com');
19
20    -- Affichage pour vérification immédiate
21    SELECT * FROM utilisateurs;
22  "
```

Le résultat de l'instruction `SELECT` doit confirmer la présence des données :

```
1  +----+-------+-------------------+
2  | id | nom   | email             |
3  +----+-------+-------------------+
4  | 1  | Alice | alice@example.com |
5  | 2  | Bob   | bob@example.com   |
6  +----+-------+-------------------+
```

3 Vérification de la Persistance des Données

L'étape suivante est de simuler la défaillance ou le remplacement du conteneur en l'arrêtant et le supprimant, puis en lançant une nouvelle instance.

Instruction 3 : Arrêt et Suppression du Conteneur Initial

Il faut arrêter et supprimer le conteneur initial.

```
1  docker stop mysql-original
2  docker rm mysql-original
```

Résultat Attendu : Bien que le conteneur ait disparu, le volume nommé `mysql_data` et son contenu sont préservés par le moteur Docker.

Instruction 4 : Lancement d'un Nouveau Conteneur

Il faut lancer un nouveau conteneur (`mysql-partage`) en le liant au même volume `mysql_data`.

```
1  docker run -d \
2    --name mysql-partage \
3    -e MYSQL_ROOT_PASSWORD=pass \
4    -v mysql_data:/var/lib/mysql \
5    mysql:8
```

Instruction 5 : Vérification de l'Accessibilité des Données

Il faut se connecter au nouveau conteneur pour vérifier si la base de données `mon_projet` et les données insérées sont toujours accessibles.

```
docker exec -it mysql-partage mysql -u root -ppass -e "
  SHOW DATABASES;
  SELECT * FROM mon_projet.utilisateurs;
"
```

Le succès de cette étape est confirmé par l'affichage de la table `utilisateurs` avec les enregistrements d'Alice et Bob, prouvant que les données sont restées persistantes et ont été partagées avec le nouveau conteneur.

4 Conclusion

L'utilisation du volume nommé `mysql_data` a permis de découpler le cycle de vie des données de celui du conteneur. Ceci est fondamental dans les environnements de production pour la mise à jour, le redémarrage ou le remplacement des services sans perte de données.