

Installation de Mariadb

1 Préalables

Note : Le SGBDR MariaDB est un *fork* (qu'est-ce que c'est ?) de **MySQL**. Grâce à sa licence libre, il est largement répandu dans les distributions Linux. Il est compatible avec **MySQL** mais ajoute de nouvelles fonctionnalités.

2 Installation

Commençons par installer le serveur **MariaDB** (sur votre serveur principal) :

```
# installation
dnf module -y install mariadb:10.11

# lancement et activation
systemctl enable --now mariadb

# sécuriser (choisir le mot de passe 'hello')
# c'est l'équivalent de mysql_secure_installation
mariadb-secure-installation
```

Le logiciel client est simplement la commande `mariadb` (l'équivalent de `mysql`) :

```
# se connecter au serveur (-p pour donner le mot de passe)
mariadb -p
```

Une fois connecté, avec les ordres ci-dessous, créez une BD, configurez les droits :

```
CREATE DATABASE moviesdb ;

GRANT ALL PRIVILEGES ON moviesdb.* TO moviesuser@localhost
IDENTIFIED BY 'moviespass' WITH GRANT OPTION;
```

Vous pouvez maintenant vous connecter avec cet utilisateur :

```
mariadb -u moviesuser -p
```

Une fois connecté

```
USE moviesdb;

CREATE TABLE test_table (id int, name varchar(50), primary key (id));

INSERT INTO test_table(id, name) values(1, "CentOS");

SELECT * FROM test_table;

SHOW TABLES ;
```

3 Deployer une application Java/SpringBoot

Téléchargez une application Spring-boot qui travaille, par défaut, avec une base de données en mémoire et testez-là (en créant un tunnel `ssh` à partir de votre poste DOSI `ssh -L9000:localhost:8081 VM`) :

```
# si cela est nécessaire
dnf -y install unzip
dnf -y install java-latest-openjdk
dnf -y install java-latest-openjdk-devel

wget https://jean-luc-massat.pedaweb.univ-amu.fr/ens/cca/spring-app.war
java -jar spring-app.war
```

Cette application utilise des paramètres par défaut qui sont définis dans le fichier `application.properties`. Vous pouvez le consulter avec les commandes ci-dessous :

Extraire et consulter les paramètres

```
## trouver le fichier de paramétrage
jar tvf spring-app.war | fgrep application.properties
## extraire le fichier de paramétrage
jar xvf spring-app.war WEB-INF/classes/application.properties
## le consulter
cat WEB-INF/classes/application.properties
## le supprimer
rm -rvf WEB-INF
```

4 Utiliser cette base de données

Nous allons maintenant lancer cette application en changeant les paramètres Spring-Boot afin d'utiliser la nouvelle base de données **MariaDB** :

Script `spring-app.sh` à préparer

```
java \  
-Dspring.datasource.driver-class-name=com.mysql.jdbc.Driver \  
-Dspring.datasource.url=jdbc:mysql://localhost/moviesdb \  
-Dspring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL8Dialect \  
-Dspring.jpa.hibernate.ddl-auto=update \  
-Dspring.datasource.username=moviesuser \  
-Dspring.datasource.password=moviespass \  
-jar spring-app.war
```

Utilisez l'application et vérifiez que les tables sont correctement créées et remplies dans la base `moviesdb`.