

L'administration d'un serveur Apache sous Linux

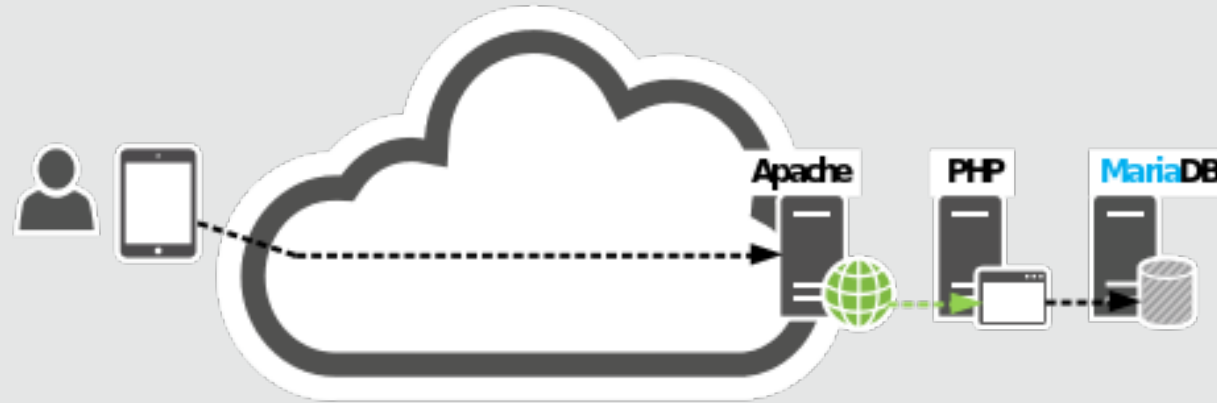
Module 12 - MariaDB



Objectifs

- Découvrir MariaDB
- Installer MariaDB
- Utiliser les requêtes SQL de base

Présentation



- Dernier élément d'une **infrastructure LAMP** : le serveur de base de données
- **MariaDB** est un **fork** du célèbre MySQL, fork initié au moment du rachat de MySQL par Sun puis Oracle

Présentation

- Fork créé en 2009, MariaDB est maintenant utilisé par exemple par [Wikipédia](#)
- Fork créé par un des développeurs de MySQL : **Michael Widenius**
- [Fonctionnement équivalent](#) à celui de MySQL
- Aucune modification à faire pour passer de MySQL à MariaDB
- Les commandes sont identiques à celles de MySQL

Installation

- Deux étapes importantes dans l'installation :

```
# apt-get install mariadb-server mariadb-client
```

```
# aptitude install mariadb-server mariadb-client
```

- Il faut ensuite sécuriser le serveur :

```
# mysql_secure_installation
```

- Définir un mot de passe pour l'administrateur et supprimer les bases et utilisateurs de test

Port d'écoute

- Par défaut, le service écoute sur le **port 3306** et sur **l'interface de loopback** (localhost)
- Si le serveur de bases de données est installé sur un autre serveur qu'Apache/PHP, il faut que le port 3306 **soit accessible sur le réseau**
- Le fichier de configuration de MariaDB est `/etc/mysql/my.cnf`

Port d'écoute

- Modifier les paramètres suivants :
 - Écoute sur l'adresse IP du réseau : `bind-address = <adresse IP>`
 - Commenter la ligne : `skip-external-locking`
 - Redémarrer MariaDB : `# systemctl restart mysql.service`

MariaDB

Installer MariaDB

Démonstration



Création d'une base de données

- Pour créer une base de données, il faut d'abord se connecter :

```
$ mysql -u root -p
```

- Puis :

```
MariaDB [ (none) ]> CREATE DATABASE mabasededonnees;
```

- Pour le moment, la base de données est une coquille vide accessible uniquement par l'administrateur

Création d'un utilisateur

- Il faut créer un utilisateur qui peut avoir les accès à la base de données :

```
MariaDB [(none)]> GRANT ALL PRIVILEGES  
ON mabasededonnee.* TO 'nom_utilisateur'@'hote'  
IDENTIFIED BY 'motdepasse';
```

```
MariaDB [(none)]> FLUSH PRIVILEGES
```

- L'utilisateur est créé avec tous les droits sur la base de données

Création d'un utilisateur

- L'hôte spécifié dans la création de l'utilisateur est la machine à partir de laquelle l'utilisateur va se connecter :
 - `localhost` : hôte local uniquement
 - Vide ou `%` : depuis n'importe quel hôte du réseau
 - `IP_SERVEUR_PHP` : l'IP du serveur sur lequel se trouve le code PHP
- Pour lister les utilisateurs :
`MariaDB [(none)]>SELECT user,host FROM mysql.user;`

Création d'un utilisateur

- La commande précédente permet de créer et d'affecter les droits à l'utilisateur

- On peut créer l'utilisateur sans droits avec accès local :

```
MariaDB [(none)]> CREATE USER 'username'@'localhost' IDENTIFIED BY 'password' ;
```

- On affecte les droits :

```
MariaDB [(none)]> GRANT ALL ON database.*  
TO 'username'@'localhost'
```

Connexion au serveur

- En administrateur du serveur : `$ mysql -u root -p`
- En utilisateur : `$ mysql -u monutilisateur -p`
- Se connecter directement à une base :
`$ mysql -u monutilisateur -p mabasededonnees`
- À chaque fois, le mot de passe est demandé en interactif

Les bases de données

- Afficher la liste des bases de données :

```
MariaDB [(none)]> SHOW DATABASE;
```

- Sélectionner une base de données :

```
MariaDB [(none)]> USE [database];
```

- Supprimer une base de données :

```
MariaDB [(none)]> DROP DATABASE [database];
```

Les tables

- Afficher la liste des tables d'une base :

```
MariaDB [database]> SHOW TABLES;
```

- Afficher la structure d'une table :

```
MariaDB [database]> DESCRIBE [table];
```

- Afficher les index :

```
MariaDB [database]> SHOW INDEX FROM [table];
```

- Supprimer une table :

```
MariaDB [database]> DROP TABLES [table];
```

Les tables

- Créer une table avec deux colonnes :

```
MariaDB [database]> CREATE TABLE [table] ([col1]  
VARCHAR(120), [col2] DATETIME);
```

- Ajouter une colonne :

```
MariaDB [database]> ALTER TABLE [table]  
ADD COLUMN [col] VARCHAR(120);
```


Les enregistrements

- Sélectionner toutes les entrées d'une table :

```
MariaDB [database]> SELECT * FROM [table];
```

- Sélectionner une entrée avec une valeur donnée :

```
MariaDB [database]> SELECT * FROM [table]  
WHERE [col] LIKE '[value]';
```

- Mise à jour d'une entrée :

```
MariaDB [database]> UPDATE [table]  
SET [col] = '[updated-value]'  
WHERE COLUMN = [value];
```

Les enregistrements

- Ajouter une entrée :

```
MariaDB [database]> INSERT INTO [table] ([col1],[col2])  
VALUES ( '[value1]', '[value2]' );
```

- Supprimer une entrée :

```
MariaDB [database]> DELETE FROM [table]  
WHERE [col] = [value];
```

Importer un script SQL

- Si on a récupéré un script SQL (dump d'une base ou script d'actions), on peut l'importer dans une base

- En local :

```
$ mysql -u login -p < fichier.sql
```

- Si le serveur SQL est distant :

```
$ mysql -u login -p -h serveursql < fichier.sql
```

Intégrer MariaDB avec PHP

- Il suffit d'ajouter les librairies PHP pour interfacer le code avec le serveur de bases de données :

```
# aptitude install php5-mysql
```

phpMyAdmin

- **phpMyAdmin** est l'outil web de référence pour gérer les bases de données :
<https://www.phpmyadmin.net/>
- **Adminer** est un peu plus basique :
<https://www.adminer.org/>
- Installation standard de phpMyAdmin :
`# aptitude install phpmyadmin`

MariaDB

Utiliser quelques commandes de base

Démonstration



Conclusion

- Vous savez maintenant installer MariaDB
- Vous connaissez les commandes de base dans MariaDB pour créer une base, un utilisateur et importer le contenu d'un fichier SQL