

Installer et configurer un serveur SGBD avec MariaDB sur Debian 8.4

Sommaire

1. Introduction

- - - - 1.1 Introduction
- - - - 1.2 Prérequis
- - - - 1.3 Version des paquets installés

2. Configuration du serveur

- - - - 2.1 Configuration du serveur

3. SGBD

- - - - 3.1 Installation
- - - - 3.2 Sécurisation

4. Services Web

- - - - 4.1 Apache2
- - - - 4.2 PHP 5
- - - - 4.3 PHPMyAdmin

Introduction

Introduction

MariaDB est un fork de MySQL permettant de mettre en place un serveur SGBD donc un **Serveur de Gestion de Base de Données**.

Prérequis

| | | | |
|-----------------|---|--|----------------|
| Nom de l'hôte | Serveur SGBD: SGBDLab | Client | Client 2 |
| Adresse IP | 172.16.0.80 | | |
| OS/ES | Debian 8.4 | Windows 7 | Debian 8.2 |
| Logiciel/Paquet | Apache2 MariaDB PHP PHPMYAdmin | Un navigateur internet: Google Chrome | Mariadb-client |

Version des paquets installés

- Apache2 – 2.4.10
- MariaDB – 10.1
- PHP5 – 5.6.13
- PHPMYAdmin – 4.2.12

Configuration du serveur

Configuration des serveur

Nous allons d'abord configurer le serveur pour que son adressage soit en IP fixe pour faciliter les connections entre les clients et le serveur (cette étape est optionnelle). Pour faire cela nous devons exécuter cette commande :

```
nano /etc/network/interfaces
```

Ensuite, chercher la ligne « *iface eth0 inet dhcp* » et remplacer celle-ci par :

```
iface eth0 inet static
```

Puis en dessous, insérer ces lignes (Remplacer les valeurs par celle de votre réseau) :

```
# Mettre une adresse IP disponible de votre réseau  
address 172.16.0.80  
# Mettre le masque de sous-réseau de votre réseau  
netmask 255.255.0.0
```

Maintenant que nous avons configuré une adresse IP fixe pour faciliter la connexion vers cette machine. Nous exécuterons la commande suivante pour permettre de rajouter le dépôt de *MariaDB* :

```
apt-get install software-properties-common
```

Pour récupérer le paquet de *MariaDB*, nous devons rajouter la signature du miroir de l'*Institut de Génétique Humaine* :

```
apt-key adv --recv-keys --keyserver keyserver.ubuntu.com 0xc9cb082a1bb943db
```

Par la suite nous rajoutons leur dépôt :

```
add-apt-repository 'deb [arch=amd64,i386] http://ftp.igh.cnrs.fr/pub/mariadb/repo/10.1/debian jessie main'
```

Enfin nous mettons le serveur à jour :

```
apt-get update
```

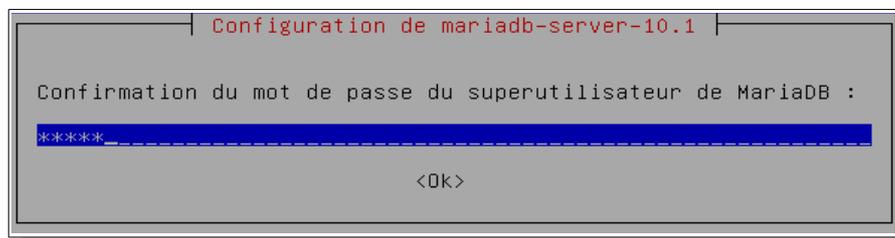
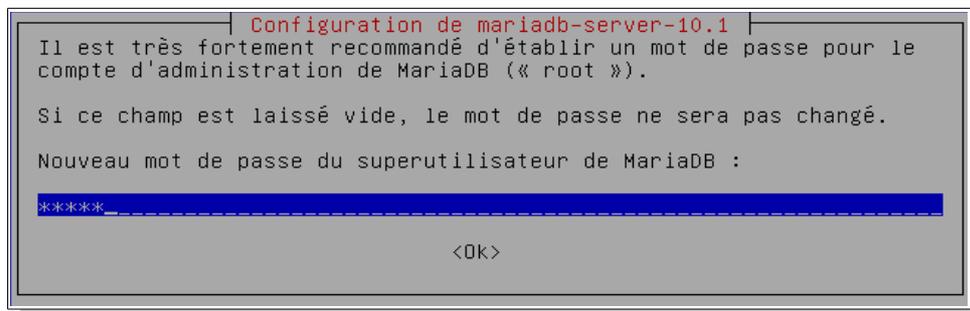
MariaDB est un SGBD open source et un fork de **MySQL** le rendant ainsi hautement compatible avec celui-ci. Il est plus performant (Entre 3% à 5%) par rapport à **MySQL** ([Benchmark](#)) et possède des améliorations, telle que la transparence sur les mises à jour de sécurité (C'est à dire que **MySQL** ne dit pas exactement les failles qui ont été corrigées, disant seulement qu'ils ont corrigé une faille, or **MariaDB** dit clairement ce qui a été corrigé), il possède plus de fonctionnalités, ... ([10 raisons de migrer vers MariaDB](#), [différence entre MariaDB et MySQL](#)).

Installation

Pour installer **MariaDB** nous exécutons cette commande :

```
apt-get install mariadb-server
```

Il nous sera demandé de valider l'installation, pour faire cela, il faut que nous appuyons sur la touche **O** puis sur **Entrer**. Il faudra ensuite choisir un mot de passe pour le service de **MariaDB** et de le confirmer.



Sécurisation

Enfin nous allons sécuriser la base de données avec cette commande :

```
mysql_secure_installation
```

La première question sur la sécurisation de la base de données est le changement de mot de passe du compte « root » avec cette fois-ci (Par rapport à celle de l'installation) des autorisations appropriées et plus sécurisé. Ici nous répondons non en appuyant sur **N** puis sur **Entrer**.

À la suite de cela une deuxième questions pour supprimer les comptes anonymes, empêchant ainsi les connexions anonymes. Nous écrivons **Y** pour répondre oui puis nous appuyons sur **Entrer** pour valider.

La troisième questions permet de désactiver les connexions distantes avec le compte « root ». Nous mettons oui en mettant **Y** puis en validant avec **Entrer**.

Puis la quatrième questions concerne la suppression de la base de données « test ». Nous répondons oui en écrivant **Y** puis en appuyant sur **Entrer**.

Enfin il nous sera demandé si nous voulons recharger les privilèges pour que les changement de droits soit pris en compte. Nous écrivons **Y** pour répondre oui puis nous appuyons sur **Entrer** pour valider.

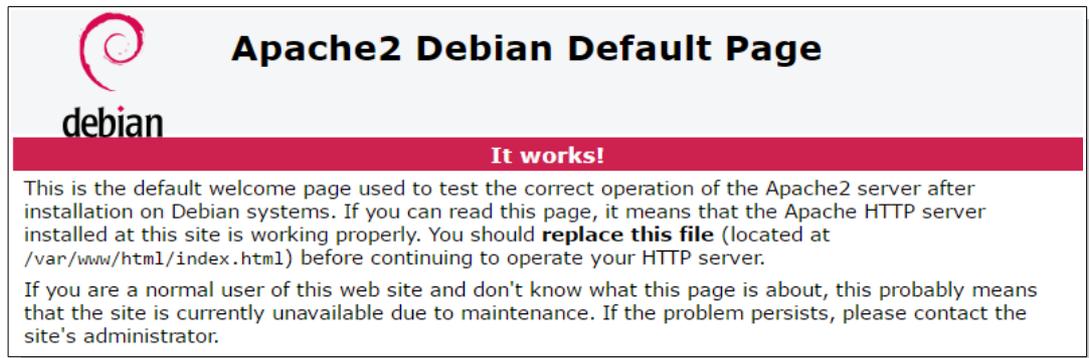
Apache2

Maintenant que la configuration du serveur est terminée, nous allons installer le service Apache2 est un service web avec le protocole HTTP permettant d'héberger un site web. Pour installer ce service il suffit d'exécuter cette commande :

```
apt-get install apache2
```

Il nous sera demandé de valider l'installation, pour faire cela, il faut que nous appuyons sur la touche **O** puis sur **Entrer**.

Une fois le service installé, nous allons vérifier si il est bien fonctionnelle, pour ce faire lancer un navigateur web puis entrer l'adresse IP du serveur dans la barre d'adresse du navigateur : <http://172.16.0.80/>, si tout ce passe bien on doit obtenir ceci :



PHP 5

PHP5 offre, grâce au *Zend Engine 2*, une gestion complète de la programmation objet. Il est nécessaire pour faire marcher *PHPMysqlAdmin* et nous le lierons à *MariaDB*. Voici la commande pour l'installation de *PHP5* :

```
apt-get install php5
```

Une fois l'installation terminée, nous installerons le module « php-mysqldb » pour le lier avec *MariaDB*, celui-ci est un remplacement de la bibliothèque « libmysqlclient » ([Introduction sur MySQLND](#)) profitant ainsi de plus de fonctionnalités ([Différence entre MySQLND et LibMySQLClient](#)) et de meilleure performance en mémoire RAM ([Benchmark](#)) pour installer ce service il suffit d'exécuter cette commande :

```
apt-get install php5-mysqldb
```

Maintenant nous allons créer le fichier « *phpinfo.php* » cela donnera beaucoup d'information sur le serveur web, en allant sur l'adresse : <http://172.16.0.80/phpinfo.php> dans le dossier « /var/www/ » pour cela, accédons dans un premier temps au dossier grâce à cette commande :

```
cd /var/www/
```

Puis créons le fichier « *phpinfo.php* » pour cela exécuter cette commande :

```
nano phpinfo.php
```

Enfin mettre ces lignes dans ce fichier :

```
<?php  
phpinfo();  
?>
```

Services Web

En allant à cette adresse : <http://172.16.0.80/phpinfo.php>, une page semblable devrait apparaître :

| PHP Version 5.6.13-0+deb8u1 | |
|---|---|
| System | Linux Febian-ProFTPD 3.16.0-4-amd64 #1 SMP Debian 3.16.7-ckt11-1+deb8u3 (2015-08-04) x86_64 |
| Build Date | Sep 7 2015 13:38:15 |
| Server API | Apache 2.0 Handler |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /etc/php5/apache2 |
| Loaded Configuration File | /etc/php5/apache2/php.ini |
| Scan this dir for additional .ini files | /etc/php5/apache2/conf.d |
| Additional .ini files parsed | /etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-gd.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mcrypt.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysqli.ini, /etc/php5/apache2/conf.d/20-readline.ini |
| PHP API | 20131106 |
| PHP Extension | 20131226 |
| Zend Extension | 220131226 |
| Zend Extension Build | API220131226.NTS |
| PHP Extension Build | API20131226.NTS |
| Debug Build | no |
| Thread Safety | disabled |
| Zend Signal Handling | disabled |
| Zend Memory Manager | enabled |
| Zend Multibyte Support | provided by mbstring |
| IPv6 Support | enabled |
| DTrace Support | enabled |
| Registered PHP Streams | https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip |
| Registered Stream Socket Transports | tcp, udp, unix, udg, ssl, sslv3, tls, tlsv1.0, tlsv1.1, tlsv1.2 |
| Registered Stream Filters | zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, mcrypt.*, mdecrypt.* |

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.6.0. Copyright (c) 1998-2015 Zend Technologies
with Zend OPcache v7.0.6-dev. Copyright (c) 1999-2015, by Zend Technologies



PHPMYAdmin

PHPMYAdmin est un service web permettant de gérer une base de donnée MySQL via un une page web réalisée en PHP. Pour l'installer il suffit d'exécuter cette commande :

```
apt-get install phpmyadmin
```

Il nous sera demandé de valider l'installation, pour faire cela, il faut que nous appuyons sur la touche **O** puis sur **Entrer**. On devra choisir le service web utiliser, ici on prendra Apache2.

```
Configuration de phpmyadmin
-----
Veuillez choisir le serveur web à reconfigurer automatiquement pour
exécuter phpMyAdmin.

Serveur web à reconfigurer automatiquement :

[ ] apache2
[ ] lighttpd

<Ok>
```

Si nous voulons configurer la base de donnée de PHPMYAdmin avec le « *dbconfig-common* », on répondra « *Oui* » il simplifiera la configuration des bases de données avec PHPMYAdmin.

```
Configuration de phpmyadmin
-----
Le paquet phpmyadmin a besoin d'une base de données installée et
configurée avant de pouvoir être utilisé. Si vous le souhaitez,
dbconfig-common peut prendre cette opération en charge.

Si vous êtes un administrateur de bases de données expérimenté et si
vous savez que vous voulez procéder à cette configuration vous-même, ou
si votre base de données est déjà installée et configurée, vous pouvez
refuser cette option. Des précisions sur la procédure se trouvent dans
/usr/share/doc/phpmyadmin.

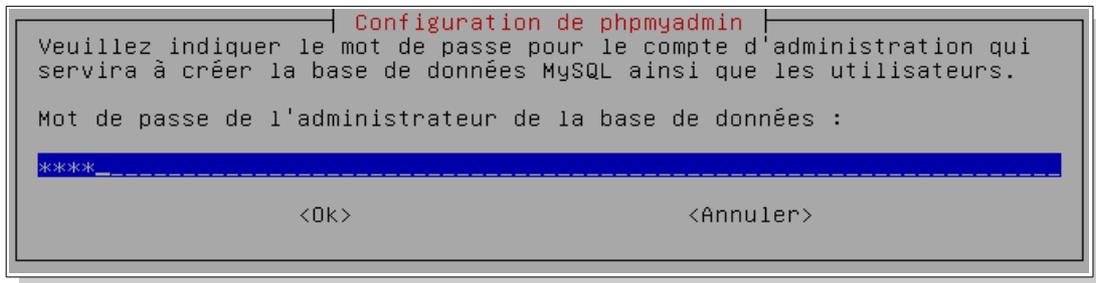
Autrement, vous devriez choisir cette option.

Faut-il configurer la base de données de phpmyadmin avec
dbconfig-common ?

<Oui> <Non>
```

Services Web

Ensuite nous devons entrer un mot de passe pour l'administrateur de la base de données :



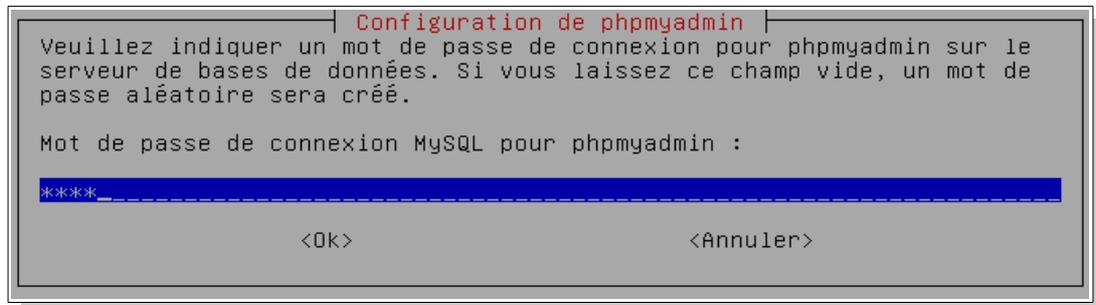
Configuration de phpmyadmin

Veillez indiquer le mot de passe pour le compte d'administration qui servira à créer la base de données MySQL ainsi que les utilisateurs.

Mot de passe de l'administrateur de la base de données :

<Ok> <Annuler>

Il sera demandé créer un mot de passe pour le service **PHPMYAdmin** et de confirmer celui ci.



Configuration de phpmyadmin

Veillez indiquer un mot de passe de connexion pour phpmyadmin sur le serveur de bases de données. Si vous laissez ce champ vide, un mot de passe aléatoire sera créé.

Mot de passe de connexion MySQL pour phpmyadmin :

<Ok> <Annuler>

Maintenant que l'installation est terminée, nous pouvons accéder à <http://172.16.0.80/phpmyadmin>, une page semblable devrait apparaître :





Bienvenue dans phpMyAdmin

Langue - Language

Français - French

Connexion

Utilisateur :

Mot de passe :