

Chapitre 5b: Installation et configuration de services

INF1070

Utilisation et administration des systèmes informatiques

Jean Privat & Alexandre Blondin Massé

Université du Québec à Montréal

v213



Plan

- 1 Installation et configuration PHP
- 2 Installation et configuration base de données - Mariadb
- 3 Paquetages XAMPP, MAMP et autres
- 4 Installation application web modulaire - Cas Moodle
- 5 Configuration des tâches planifiées

Installation et configuration PHP

Installation et version PHP

- <https://www.php.net/manual/en/install.php>
- Choisir une version et au besoin installer module apache

```
$ sudo apt-cache policy php # voir version par défaut
$ sudo apt-get install php7.2-common php7.2-cli
$ sudo apt-get install libapache2-mod-php7.2
$ php -version # verifier version après installation
$ sudo apt-get install php7.2-mbstring # install module
$ sudo phpenmod -v 7.2 mbstring
# phpenmod pour activer et phpdismod pour desactiver
$ sudo apt-get install php7.2-intl # internationalisation
$ sudo phpenmod -v 7.2 intl
```

Configuration PHP et intégration avec Apache

Fichier de configuration

`php.ini`

- Pour localiser le fichier:

```
$ php --ini
```

- Documentation `php.ini`

<https://www.php.net/manual/en/configuration.file.php>

Intégration module dans apache

- activer(`a2enmod`)/ désactiver (`a2dismod`) module php

```
$ sudo a2enmod php7.2 # activer module php7.2
```

```
$ sudo service apache2 restart # redemarrer apache
```

Vérification la configuration php

- Ligne de commande

```
$ php -i
```

- Page web

```
<?php //fichier phpinfo.php  
phpinfo();  
>
```

- Mettre phpinfo.php dans /var/www/html ou dossier accessible par web
- Visiter la page <http://127.0.0.1/phpinfo.php>

Taille maximale fichier dans formulaire PHP

- Ajuster les paramètres suivants dans `php.ini`

```
memory_limit = 128M
upload_max_filesize = 2M
post_max_size = 8M
```

- Pour mettre 2 GB par exemple, écrire **2G**
- Parfois, ça peut-être nécessaire de redemarrer ou recharger le serveur apache après modification de `php.ini`
- Plus d'info: <https://www.php.net/manual/fr/ini.core.php>

Installation et configuration base de données - Mariadb

Installation Mariadb/MySQL

- <https://linuxize.com/post/how-to-install-mariadb-on-ubuntu-18-04/>

```
$ sudo apt-get update
$ sudo apt-get install mariadb-server mariadb-client
$ /etc/init.d/mysql start # démarrage
$ mysql -u root # pour tester connexion
# mot de passe par défaut dans /etc/mysql/debian.cnf
```

- Pour Mysql: <https://doc.ubuntu-fr.org/mysql>
- Installation module php pour programme PHP

```
$ sudo apt-get install php7.2-mysql php-mdb2-driver-mysql
```

- Décommenter module dans `php.ini`

```
extension=pdo_mysql
extension=mysqli
```

Configuration Mariadb/MySQL

```
/etc/mysql/my.cnf #  
/etc/mysql/mariadb.cnf # configuration globale mariadb  
/etc/mysql/debian.cnf # infos socket et root
```

Moteur de stockage InnoDB permet

- Transactions ACID (atomiques, cohérentes, isolées et durables)
- Gestion des clés étrangères (avec vérification de la cohérence)

<https://fr.wikipedia.org/wiki/InnoDB>, (détails aux cours BD).

UTF8 dans le fichier my.conf

```
# Dans les sections [client] et [mysql]  
default-character-set=utf8mb4  
# Dans la section [mysqld]  
collation-server = utf8mb4_unicode_ci  
init-connect='SET NAMES utf8mb4'  
character-set-server = utf8mb4
```

<https://mariadb.com/kb/en/library/setting-character-sets-and-collations/#example-changing-the-default-character-set-to-utf-8>

Créer utilisateur avec privilèges administrateur

```
$ mysql -u root
MariaDB [(none)]> use mysql
MariaDB [mysql]> create user 'ab123456'@'localhost' \
    -> identified by 'motpasse';
MariaDB [mysql]> grant all privileges on *.* \
    -> to 'ab123456'@'localhost';
MariaDB [mysql]> exit # pour quitter
$ mysql -u ab123456 -p # se connecter avec mot de passe
```

Installation et configuration phpMyAdmin

- Administration MySql/MariaDB par interface Web

Installation

```
$ sudo apt-get install phpmyadmin
```

Configuration (si pas fait à l'installation)

```
$ sudo dpkg-reconfigure phpmyadmin  
$ sudo dpkg-reconfigure dbconfig-common
```

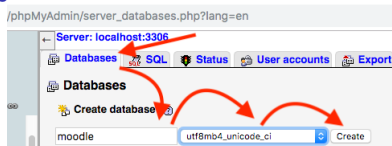
Visiter: <http://127.0.0.1/phpmyadmin>

Documentation et demo

- <https://www.phpmyadmin.net/>
- <https://demo.phpmyadmin.net/master-config/>

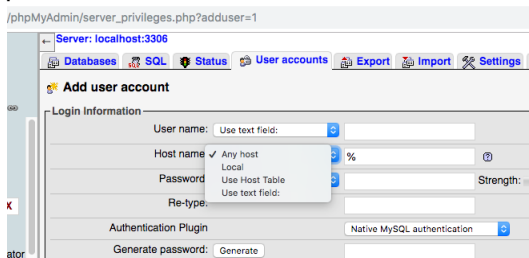
Administration base de données

Création BD et jeux de caractères



Création utilisateur

- Accès à partir du serveur: localhost
- Accès de partout



Permissions globales SGBD

- Permissions globales dans le Système de Gestion des Bases de Données (SGBD)

MyAdmin/server_privileges.php?adduser=1

Server: localhost:3306

Databases SQL Status User accounts Export Import Settings Replication Variables

Database for user account

Create database with same name and grant all privileges.
 Grant all privileges on wildcard name (username_%).

Global privileges Check all

Note: MySQL privilege names are expressed in English.

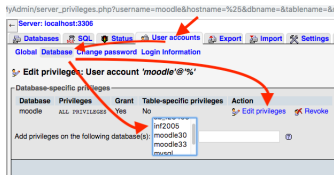
<input type="checkbox"/> Data	<input type="checkbox"/> Structure	<input type="checkbox"/> Administration	<input type="checkbox"/> Resource limits
<input type="checkbox"/> SELECT	<input type="checkbox"/> CREATE	<input type="checkbox"/> GRANT	Note: Setting these options to 0 (zero) removes the limit.
<input type="checkbox"/> INSERT	<input type="checkbox"/> ALTER	<input type="checkbox"/> SUPER	MAX QUERIES PER HOUR <input type="text" value="0"/>
<input type="checkbox"/> UPDATE	<input type="checkbox"/> INDEX	<input type="checkbox"/> PROCESS	MAX UPDATES PER HOUR <input type="text" value="0"/>
<input type="checkbox"/> DELETE	<input type="checkbox"/> DROP	<input type="checkbox"/> RELOAD	MAX CONNECTIONS PER HOUR <input type="text" value="0"/>
<input type="checkbox"/> FILE	<input type="checkbox"/> CREATE TEMPORARY TABLES	<input type="checkbox"/> SHUTDOWN	MAX USER_CONNECTIONS <input type="text" value="0"/>
	<input type="checkbox"/> SHOW VIEW	<input type="checkbox"/> SHOW DATABASES	
	<input type="checkbox"/> CREATE ROUTINE	<input type="checkbox"/> LOCK TABLES	
	<input type="checkbox"/> ALTER ROUTINE	<input type="checkbox"/> REFERENCES	
	<input type="checkbox"/> EXECUTE	<input type="checkbox"/> REPLICATION CLIENT	
	<input type="checkbox"/> CREATE VIEW	<input type="checkbox"/> REPLICATION SLAVE	
	<input type="checkbox"/> EVENT	<input type="checkbox"/> CREATE USER	
	<input type="checkbox"/> TRIGGER		

SSL

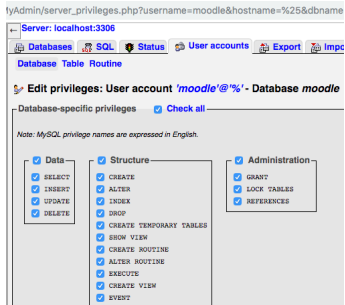
REQUIRE NONE
 REQUIRE SSL
 REQUIRE X509

Permissions dans la bases de données

- Ajouter permissions base de données



- Modifier permissions base de données



Utilisateur BD Vs Application

Utilisateur BD pour une application

- Souvent un seul utilisateur pour toute l'application.
- On mets les informations dans le fichier de configuration de l'application
- Dans la table des utilisateurs du SGBD

Utilisateurs d'une application

- Souvent les applications on une table spéciale pour les utilisateurs
- Ne peut pas se connecter directement dans la BD
- Il peut y avoir des milliers des comptes. Ex: comptes des étudiants sur moodle
- Parfois, on utilise des applications spécialisées pour gérer ces comptes. EX: LDAP/AD

Paquetages XAMPP, MAMP et autres

XAMPP ou LAMPP

- Connus sous LAMP
- Paquetage Apache, MariaDB, PHP et perl
- Contient aussi phpMyAdmin
- Préconfiguré comme une application
- MacOS, Linux, Windows
- Pour le développement et non la production
- Au besoin, ajuster la configuration avant d'utiliser en production
- Le tout s'installe dans un dossier
- <https://www.apachefriends.org/fr/index.html>

MAMP et autres paquetages

MAMP

- Paquetage Apache, Nginx, MariaDb, PHP
- MacOS, Windows
- Préconfiguré comme une application
- Pour le développement
- Contient aussi phpMyAdmin
- Le tout s'installe dans un dossier
- il y a une version Pro (MAMP PRO)
- <https://www.mamp.info/en/>

Autres paquetages

- <http://www.wampserver.com/>
- <https://www.easyphp.org/>
- <http://devilbox.org/>

Installation application web modulaire - Cas Moodle

Principe général

Une application web modulaire comporte un noyau et des plugins

Noyau (core) et Modules

Quelques applications

- Drupal (CMS) <https://www.drupal.org/>
- Prestashop (E-commerce) <https://www.prestashop.com>
- Moodle (LMS) <https://moodle.org/>
- WordPress (CMS) <https://wordpress.org/>

Les environnements de développement

Chaque environnement est une instance de l'application qui a un rôle bien spécifique.

Dev

- Environnement sur lequel les développeurs travaillent.
- Instable

Test

- Permet de tester les fonctionnalités
- Le code d'une version de l'application
- Accessible aux testeurs

Preprod

- Application et données représentatives (ou copie prod)
- Souvent chez le client

Prod

- L'application réellement utilisée en production

Moodle - Code source et configuration

Code source et documentation

- <https://github.com/moodle/moodle.git>
- <https://moodle.org/>

Configuration

- Ne pas mettre le dans le dépôt de code source.

```
config.php # fichier de configuration  
config-dist.php # exemple distribué avec le code source
```

Contient:

- Information de la base de donnée (nom BD, serveur, utilisateur, mot de passe)
- Chemins vers les dossiers de données
- url de l'application

Moodle - Installation

Installation et exigences systèmes

- Créer une base de données et un utilisateur
- Copier moodle dans dossier web /var/www/html/moodle
- Copier config-dist.php vers config.php et modifier

```
$CFG->dbtype      = 'mariadb';  
$CFG->dbhost      = 'localhost';  
$CFG->dbname      = 'moodle'; # ajuster  
$CFG->dbuser      = 'username'; # ajuster  
$CFG->dbpass      = 'password'; # ajuster  
$CFG->wwwroot     = 'http://localhost/moodle';  
$CFG->dataroot    = '/var/moodledata'; # fichiers
```

- Installation par l'interface web

Visiter <http://localhost/moodle>

- Satisfaire les exigences systèmes et continuer

Moodle - Ajout des plugins

- plugins <https://moodle.org/plugins/>
- Satisfaire exigences avant d'installer un autre plugin.
- Suivre la procédure expliquée pour le plugin

Ajout par l'interface web

Visiter <https://localhost/moodle/admin/plugins.php>

Ajout plugins dans les dossiers

```
$ php admin/cli/upgrade.php # dans le dossier moodle
```

Ou visiter <https://localhost/moodle/admin/index.php>

Moodle - gestion des utilisateurs

Administrateurs

- `admin`

Autres utilisateurs

- Manuellement
- Méthode authentification
<https://docs.moodle.org/37/en/Authentication>

Les rôles et permissions

- administrateurs, manageurs, professeur, professeur non-editeur
- étudiant, invité (guest), etc
- autres rôles définis

Demo

- <https://moodle.org/demo/>

Configuration des tâches planifiées

Tâches planifiés Crontab – Principe

```
$ sudo apt-get install cron # installation si besoin
# crontab [ -u user ] [ -i ] { -e | -l | -r }
# Si pas -u alors utilisateur courant
# -r pour supprimer le cron de l'utilisateur
$ crontab -l # voir ses taches planifiées
$ crontab -e # Pour editer
```

Format de cron <https://help.ubuntu.com/community/CronHowto>

```
01 * 1 1 * commande
# min heure jour mois jourDeLaSemaine commande
# min (0-59), heure (0-23), jour (1-31)
# mois (1-12), jourDeLaSemaine (0-7, 0=7=dimanche)
12,21,38,52 * 1 1 * commande
30 */2 * * * commande # aux 30min toutes les 2 heures
# * signifie pour chaque instance
# , sépare les valeurs précises
# / pour les intervalles (* / 6 = tous les 6 ...)
```

- Plus info et exemples <https://crontab.guru/>

Services et tâches d'une application - Moodle

- Permettent d'automatiser l'exécution des tâches

Principale tâche planifiée

<https://docs.moodle.org/37/en/Cron>

```
*/5 * * * * php /var/www/html/moodle/admin/cli/cron.php
```

- Exécute les tâches planifiées de l'application
- Programmée au niveau système

Autres tâches planifiées

- Exécutées par le cron principal
- Programmées au niveau de l'application

Visiter <http://localhost/moodle/admin/tool/task/scheduledtasks.php>