

# Installation de GLPI sur Debian 12



<b>I. Prérequis</b>	<b>3</b>
<b>II. Préparer le serveur pour installer GLPI</b>	<b>4</b>
A. Installer le socle LAMP	4
B. Préparer une base de données pour GLPI	4
C. Télécharger GLPI et préparer son installation	6
D. Préparer la configuration Apache2	7
E. Utilisation de PHP8.2-FPM avec Apache2	8
<b>III. Installation de GLPI</b>	<b>9</b>

# I. Prérequis

Mise à jour des paquets

Machine Debian avec IP fixe

## II. Préparer le serveur pour installer GLPI

### A. Installer le socle LAMP

```
sio@debian:~$ sudo apt-get install apache2 php mariadb-server
```

Installer toutes les extensions nécessaires au bon fonctionnement de GLPI

```
sio@debian:~$ sudo apt-get install php-xml php-common php-json php-mysql php-mbstring  
php-curl php-gd php-intl php-zip php-bz2 php-imap php-apcu
```

Si vous envisagez d'associer GLPI avec un annuaire LDAP comme l'Active Directory

```
sio@debian:~$ sudo apt-get install php-ldap
```

### B. Préparer une base de données pour GLPI

Sécurisation de MariaDB

```
sio@debian:~$ sudo mysql_secure_installation
```

Suivre l'exemple ci-dessous pour chaque option

```
Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] n
... skipping.

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
```

Se connecter à MariaDB

```
sio@debian:~$ sudo mysql -u root -p
```

Création de la base de données ainsi que l'utilisateur

```
MariaDB [(none)]> create database db_glpi;
Query OK, 1 row affected (0,000 sec)

MariaDB [(none)]> grant all privileges on db_glpi.* to glpi_adm@localhost id
entified by "Sio2019+";
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0,000 sec)

MariaDB [(none)]> exit
Bye
sio@debian:~$ █
```

## C. Télécharger GLPI et préparer son installation

Télécharger l'archive GLPI

```
sio@debian:~$ cd /tmp
sio@debian:/tmp$ wget https://github.com/glpi-project/glpi/releases/download/10.0.10/glpi-10.0.10.tgz
```

Décompression de l'archive dans un autre répertoire

```
sio@debian:/tmp$ sudo tar -xzf glpi-10.0.10.tgz -C /var/www/
```

Définir un utilisateur "www-data" propriétaire sur les fichiers GLPI

```
sio@debian:/tmp$ sudo chown www-data /var/www/glpi/ -R
```

Créer le répertoire "/etc/glpi" qui va recevoir les fichiers de configuration de GLPI. Donner des autorisations à www-data sur ce répertoire pour pouvoir y accéder

```
sio@debian:/tmp$ sudo mkdir /etc/glpi
sio@debian:/tmp$ sudo chown www-data /etc/glpi/
```

Déplacer le répertoire "config" de GLPI vers ce dossier

```
sio@debian:/tmp$ sudo mv /var/www/glpi/config /etc/glpi
```

Même chose avec répertoire "/var/lib/glpi"

```
sio@debian:/tmp$ sudo mkdir /var/lib/glpi
sio@debian:/tmp$ sudo chown www-data /var/lib/glpi/
```

Déplacer le dossier "files" qui contient la majorité des fichiers de GLPI : CSS, plugins, etc.

```
sio@debian:/tmp$ sudo mv /var/www/glpi/files /var/lib/glpi
```

Créer le répertoire "/var/log/glpi" destiné à stocker les journaux de GLPI

```
sio@debian:/tmp$ sudo mkdir /var/log/glpi
sio@debian:/tmp$ sudo chown www-data /var/log/glpi
```

Création des fichiers de configuration

```
sio@debian:/tmp$ sudo nano /var/www/glpi/inc/downstream.php
```

Le contenu ci-dessous indique le chemin vers le répertoire de configuration

```
GNU nano 7.2 /var/www/glpi/inc/downstream.php *
<?php
define('GLPI_CONFIG_DIR', '/etc/glpi/');
if (file_exists(GLPI_CONFIG_DIR . '/local_define.php')) {
    require_once GLPI_CONFIG_DIR . '/local_define.php';
}
```

Création d'un autre fichier de configuration

```
sio@debian:/tmp$ sudo nano /etc/glpi/local_define.php
```

Le contenu ci-dessous permet de déclarer deux variables permettant de préciser les chemins vers les répertoires "files" et "log" que préparés précédemment

```
GNU nano 7.2 /etc/glpi/local_define.php *
<?php
define('GLPI_VAR_DIR', '/var/lib/glpi/files');
define('GLPI_LOG_DIR', '/var/log/glpi');
```

## D. Préparer la configuration Apache2

Créer un nouveau fichier de configuration qui va permettre de configurer le VirtualHost dédié à GLPI

```
sio@debian:/tmp$ sudo nano /etc/apache2/sites-available/support.sio.local.conf
```

```
GNU nano 7.2 /etc/apache2/sites-available/support.sio.local.conf *
<VirtualHost *:80>
  ServerName support.sio.local

  DocumentRoot /var/www/glpi/public

  # If you want to place GLPI in a subfolder of your site (e.g. your virtual host is serving multiple applications),
  # you can use an Alias directive. If you do this, the DocumentRoot directive MUST NOT target the GLPI directory itself.
  # Alias "/glpi" "/var/www/glpi/public"

  <Directory /var/www/glpi/public>
    Require all granted

    RewriteEngine On

    # Redirect all requests to GLPI router, unless file exists.
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteRule ^(.*)$ index.php [QSA,L]
  </Directory>
</VirtualHost>
```

Activer ce nouveau site dans Apache2

```
sio@debian:/tmp$ sudo a2ensite support.sio.local.conf
```

Désactiver le site par défaut car il est inutile

```
sio@debian:/tmp$ sudo a2dissite 000-default.conf
```

Activer le module "rewrite"

```
sio@debian:/tmp$ sudo a2enmod rewrite
```

Redémarrer le service Apache2

```
sio@debian:/tmp$ sudo systemctl restart apache2
```

## E. Utilisation de PHP8.2-FPM avec Apache2

Pour utiliser PHP en tant que moteur de scripts avec Apache2, il est recommandé d'utiliser PHP-FPM

Installer PHP8.2-FPM

```
sio@debian:/tmp$ sudo apt-get install php8.2-fpm
```

Activer deux modules dans Apache et la configuration de PHP-FPM

```
sio@debian:/tmp$ sudo a2enmod proxy_fcgi setenvif
```

```
sio@debian:/tmp$ sudo a2enconf php8.2-fpm
```

```
sio@debian:/tmp$ sudo systemctl reload apache2
```

Editer le fichier "/etc/php/8.2/fpm/php.ini"

```
sio@debian:/tmp$ sudo nano /etc/php/8.2/fpm/php.ini
```

Chercher l'option session.cookie\_httponly et mettre la valeur "on"

```
; Whether or not to add the httpOnly flag to the cookie, which makes it
; inaccessible to browser scripting languages such as JavaScript.
; https://php.net/session.cookie-httponly
session.cookie_httponly = on
```

Redémarrer PHP-FPM

```
sio@debian:/tmp$ sudo systemctl restart php8.2-fpm.service
```

Modifier VirtualHost pour préciser à Apache2 que PHP-FPM doit être utilisé pour les fichiers PHP

```
GNU nano 7.2 /etc/apache2/sites-available/support.sio.local.conf *
```

```
<VirtualHost *:80>
    ServerName support.sio.local

    DocumentRoot /var/www/glpi/public

    # If you want to place GLPI in a subfolder of your site (e.g. your virt
    # you can use an Alias directive. If you do this, the DocumentRoot dire
    # Alias "/glpi" "/var/www/glpi/public"

    <Directory /var/www/glpi/public>
        Require all granted

        RewriteEngine On

        # Redirect all requests to GLPI router, unless file exists.
        RewriteCond %{REQUEST_FILENAME} !-f
        RewriteRule ^(.*)$ index.php [QSA,L]
    </Directory>

    <FilesMatch \.php$>
        SetHandler "proxy:unix:/run/php/php8.2-fpm.sock|fcgi://localhost/"
    </FilesMatch>

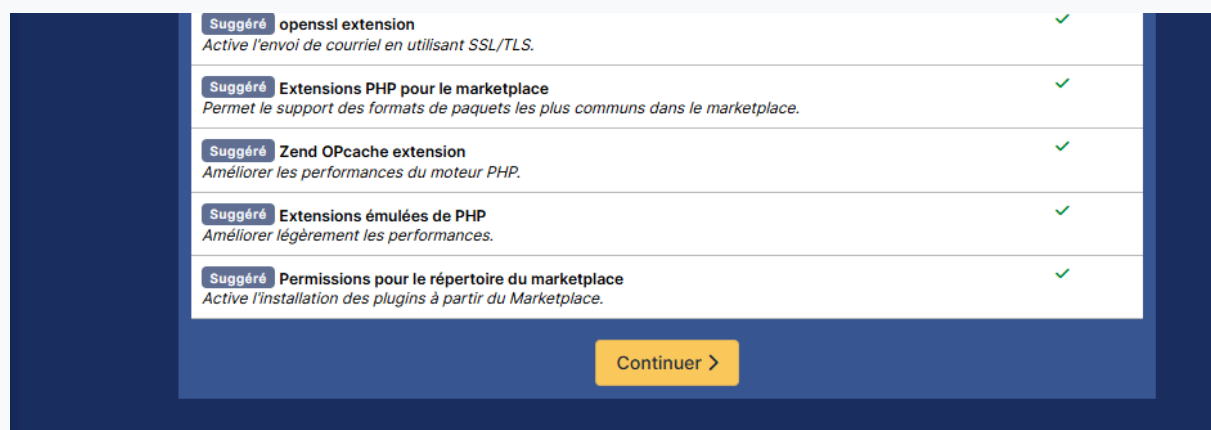
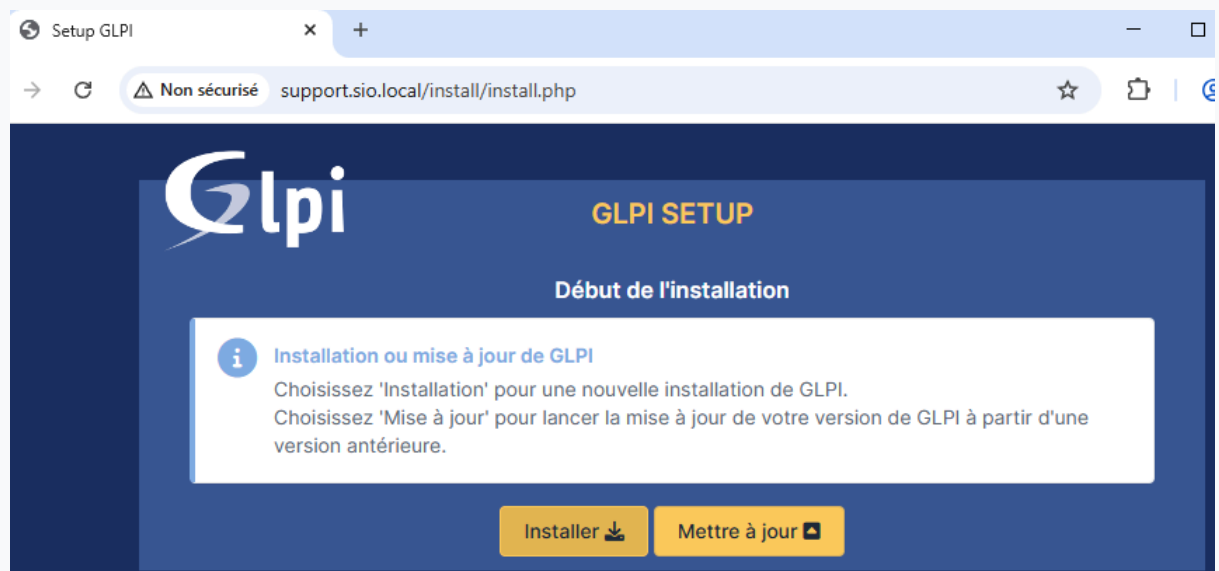
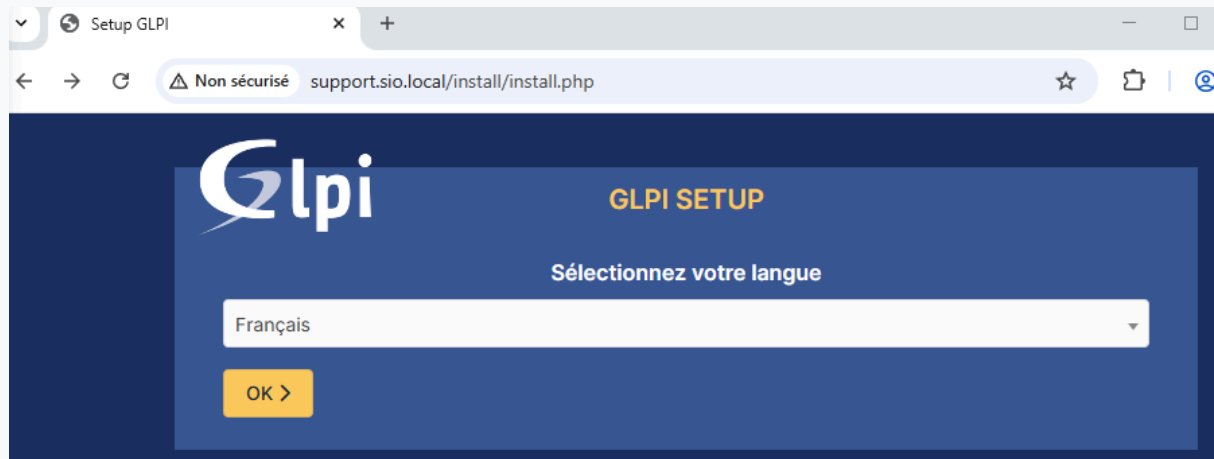
</VirtualHost>
```

Relancer Apache2

```
sio@debian:/tmp$ sudo systemctl restart apache2
```



### III. Installation de GLPI





## GLPI SETUP

### Étape 1

#### Configuration de la connexion à la base de données

Serveur SQL (MariaDB ou MySQL)

Utilisateur SQL

Mot de passe SQL

Continuer >



## GLPI SETUP

### Étape 2

#### Test de connexion à la base de données

✓ Connexion à la base de données réussie

Veuillez sélectionner une base de données :

Créer une nouvelle base ou utiliser une base existante :



db\_glpi

Continuer >



## GLPI SETUP


### Étape 6

### L'installation est terminée

Les identifiants et mots de passe par défaut sont :

- glpi/glpi pour le compte administrateur
- tech/tech pour le compte technicien
- normal/normal pour le compte normal
- post-only/postonly pour le compte postonly

Vous pouvez supprimer ou modifier ces comptes ainsi que les données initiales.

 Utiliser GLPI



### Connexion à votre compte

Identifiant

glpi

Mot de passe

....

Source de connexion

Base interne GLPI

☒ Se souvenir de moi

Se connecter

