

Mise en de GLPI 11

Mise en place de GLPI 11.0.4 sous Debian 13

- [a\) Installation des composant requis :](#)
- [b\) Création de la base de données :](#)
- [c\) Téléchargement et installation de GLPI :](#)
- [d\) Configuration de GLPI](#)
- [e\) Sécurisation de GLPI](#)

a) Installation des composant requis :

a) Installation des composant requis :

Dans un premier temps on installe les composant de base :

Apache2 pour le serveur web et mariadb-server pour le serveur de base de données

```
sudo apt-get install apache2 php8.4-fpm mariadb-server
```

```
root@GLPI-VM:/home/glpi# sudo apt-get install apache2 php mariadb-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
 apache2-bin apache2-data apache2-utils galera-4 gawk libapache2-mod-php8.4 libapr1t64 libaprutil1-dbd-sqlite3
 libaprutil1-ldap libaprutil1t64 libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl libdbd-mariadb-perl
 libdbi-perl libfcgi-bin libfcgi-perl libfcgi0t64 libhtml-template-perl liblua5.4-0 libmariadb3 libpcre2-posix3
 libsigsegv2 libterm-readkey-perl mariadb-client mariadb-client-core mariadb-common mariadb-plugin-provider-bzip2
 mariadb-plugin-provider-lz4 mariadb-plugin-provider-lzma mariadb-plugin-provider-lzo mariadb-plugin-provider-snappy
 mariadb-server-core mysql-common php-common php8.4 php8.4-cli php8.4-common php8.4-opcache php8.4-readline pv rsync
 socat
Paquets suggérés :
 apache2-doc apache2-suexec-pristine | apache2-suexec-custom ufw gawk-doc php-pear libmldbm-perl libnet-daemon-perl
 libsql-statement-perl libipc-sharedcache-perl mailx mariadb-test netcat-openbsd doc-base python3-braceexpand
Les NOUVEAUX paquets suivants seront installés :
 apache2 apache2-bin apache2-data apache2-utils galera-4 gawk libapache2-mod-php8.4 libapr1t64
 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl
 libdbd-mariadb-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0t64 libhtml-template-perl liblua5.4-0 libmariadb3
 libpcre2-posix3 libsigsegv2 libterm-readkey-perl mariadb-client mariadb-client-core mariadb-common
 mariadb-plugin-provider-bzip2 mariadb-plugin-provider-lz4 mariadb-plugin-provider-lzma mariadb-plugin-provider-lzo
 mariadb-plugin-provider-snappy mariadb-server mariadb-server-core mysql-common php-common php8.4 php8.4-cli
 php8.4-common php8.4-opcache php8.4-readline pv rsync socat
0 mis à jour, 46 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 27,6 MB dans les archives.
```

Après on installe les composant pour l'exécution de glpi (pour GLPI ce sont des modules PHP) :

```
sudo apt install php8.4-{curl,gd,intl,mysql,zip,bcmath,mbstring,xml,bz2,ldap}
```

```
Installation de :
  php8.4-bcmath  php8.4-curl  php8.4-intl  php8.4-mbstring  php8.4-xml
  php8.4-bz2     php8.4-gd    php8.4-ldap  php8.4-mysql     php8.4-zip

Installation de dépendances :
  libonig5  libzip5

Sommaire :
  Mise à niveau de : 0. Installation de : 12Supprimé : 0. Non mis à jour : 0
Taille du téléchargement : 1 699 kB
  Espace nécessaire : 6 506 kB / 56,7 GB disponible

Continuer ? [0/n] o|
```

Puis configuration de mysql pour la base de données :

```
sudo mysql_secure_installation
```

```
gtpi@GLPI-VH:~$ sudo mariadb-secure-installation
```

```
NOTE: MariaDB is secure by default in Debian. Running this script is  
useless at best, and misleading at worst. This script will be  
removed in a future MariaDB release in Debian. Please read  
mariadb-server.README.Debian for details.
```

```
Enter root user password or leave blank:
```

```
Enter current password for root (enter for none):  
OK, successfully used password, moving on ...
```

```
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!
```



On change le mot de
passe root, dans la
simplicité je remets le
même de debian

```
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.
```

On interdit les utilisateur anonyme

```
Remove anonymous users? [Y/n] y
```

```
SQL executed without errors!  
The operation might have been successful, or it might have not done anything.
```



```
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
```

```
SQL executed without errors!  
The operation might have been successful, or it might have not done anything.
```



On interdit à root de se connecté
à distance

```
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 ...  
SQL executed without errors!  
The operation might have been successful, or it might have not done anything.  
- Removing privileges on test database ...  
SQL executed without errors!  
The operation might have been successful, or it might have not done anything.
```



On enlève les base
de donnée d'exemple

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

```
Reload privilege tables now? [Y/n] y
```

```
... Success!
```



On applique toutes
les modifs au table

```
Cleaning up ...
```

```
All done! If you've completed all of the above steps, your MariaDB
```

b) Création de la base de données :

b) Création de la base de données :

Puis connexion à mariadb pour crée la base de données :

```
sudo mysql -u root -p
```

Puis on crée la base de donnée avec le mdp et un utilisateur qui à le droit de l'utilisais pour après le setup sur glpi plus tard. Donc oublié pas le mot de passe :

```
CREATE DATABASE db23_glpi; GRANT ALL PRIVILEGES ON db23_glpi.* TO glpi_admin@localhost  
IDENTIFIED BY "MotDePasseRobuste"; FLUSH PRIVILEGES; EXIT;
```

```
MariaDB [(none)]> CREATE DATABASE db23_glpi; GRANT ALL PRIVILEGES ON db23_glpi.* TO  
glpi_admin@localhost IDENTIFIED BY "MotDePasseRobuste"; FLUSH PRIVILEGES; EXIT  
Query OK, 1 row affected (0,028 sec)  
  
Query OK, 0 rows affected (0,099 sec)  
  
Query OK, 0 rows affected (0,010 sec)
```



```
sudo mkdir /etc/glpi
sudo chown www-data /etc/glpi/
sudo mv /var/www/html/glpi/config /etc/glpi
```

```
root@GLPI-VM:/tmp# sudo mkdir /etc/glpi
sudo chown www-data /etc/glpi/
sudo mv /var/www/html/glpi/files /var/lib/glpi
```

Maintenant, passons au dossier **lib**, ce qui contient par exemple les **plugins**, des **CSS**, etc. On fait également comme au dessus :

```
sudo mkdir /var/lib/glpi
sudo chown www-data /var/lib/glpi/
sudo mv /var/www/html/glpi/files /var/lib/glpi
```

```
root@GLPI-VM:/tmp# sudo mkdir /var/lib/glpi
sudo chown www-data /var/lib/glpi/
sudo mv /var/www/html/glpi/files /var/lib/glpi
```

Finissons par les **logs** mais comme nous venons d'installer GLPI, nous n'allons rien déplacer, juste créer et changer le propriétaire du dossier :

```
sudo mkdir /var/log/glpi
sudo chown www-data /var/log/glpi
```

Dans le second temps, on va créer des fichiers **PHP** dans GLPI pour changer les liens des différents dossiers que nous avons modifiés précédemment.

On va en premier créer un premier fichier pour définir le dossier "**config**" et des **variable** qui nous servira dans un second temps :

```
sudo nano /var/www/html/glpi/inc/downstream.php
```

```
GNU nano 8.4 /var/www/html/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';
}
```

Puis on va créer un second fichier php pour permettre de donner des variables. Pour nous ici, se sera que pour les dossiers **files** et **log** mais on peut en mettre bien d'autres :

```
sudo nano /etc/glpi/local_define.php
```

```
GNU nano 8.4 /etc/glpi/local_define.php *
<?php
define('GLPI_VAR_DIR', '/var/lib/glpi/files');
define('GLPI_LOG_DIR', '/var/log/glpi');
```

Nous avons fini avec le déplacement de fichier, nous passons maintenant à la configuration web avec la création d'un fichier de configuration pour **apache2** :

```
sudo nano /etc/apache2/sites-available/glpi.conf
```

```
<VirtualHost *:80>
    ServerName glpi.lerenard.eu
    DocumentRoot /var/www/html/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 its>
    # Alias "/glpi" "/var/www/glpi/public"

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

        RewriteEngine On

        # Ensure authorization headers are passed to PHP.
        # Some Apache configurations may filter them and break usage of API, CalDAV, ...
        RewriteCond %{HTTP:Authorization} ^(.+)$
        RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]

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

    </Directory>
</VirtualHost>
```

```

<VirtualHost *:80>
  ServerName glpi.lerenard.eu
  DocumentRoot /var/www/html/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 its
  # Alias "/glpi" "/var/www/glpi/public"

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

    RewriteEngine On

    # Ensure authorization headers are passed to PHP.
    # Some Apache configurations may filter them and break usage of API, CalDAV, ...
    RewriteCond %{HTTP:Authorization} ^(.+)$
    RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]

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

  </Directory>
</VirtualHost>

```

Au début il y a "**ServerName**", vous pouvez mettre ce que vous voulez, mais c'est recommandé de mettre le même nom de domaine qu'il y a en interne (si vous avez un nom de domaine en local), si vous désirez de mettre un certificat TLS/SSL pour l'https par exemple.

Configuration de apache2 pour dès que vous tapé l'ip sur un navigateur, il fasse la redirection vers GLPI et activation de PHP8.4-fpm pour l'exécution plus optimisé de script (et comme on en a fait au dessus) :

```

sudo a2ensite glpi.conf #(permet d'activé le fichier conf créé au-dessus pour glpi sur apache)
sudo a2dissite 000-default.conf #(permet de désactivé le fichier 000-default.conf qui fais
référence à index.html)
sudo a2enmod rewrite #(permet l'activation du module rewrite pour apache2)
sudo a2enmod proxy_fcgi setenvif
sudo a2enconf php8.4-fpm
sudo systemctl restart apache2 #(redémarrage d'apache2)

```

d) Configuration de GLPI

d) Configuration de GLPI :

Donc nous sélectionnons français :



Comme on veut installer GLPI on fait installer (Mettre à jour si une autre version de GLPI était déjà installé):



Si vous avez bien effectué les commandes au-dessus vous pourrez faire continuer sinon retourné au début :

Étape 0

Vérification de la compatibilité de votre environnement avec l'exécution de GLPI

TESTS EFFECTUÉS	RÉSULTATS
Requis Paraeer PHP	✓
Requis Taille d'entier maximal de PHP <i>Le support des entiers 64 bits est nécessaire pour les opérations relatives aux adresses IP (inventaire réseau, filtrage des clients API, ...).</i>	✓
Requis Configuration des sessions	✓
Requis Mémoire allouée	✓
Requis Extensions du noyau de PHP	✓
Requis mysql extension <i>Requis pour l'accès à la base de données.</i>	✓
Requis curl extension <i>Requis pour l'accès à distance aux ressources (requêtes des agents d'inventaire, Marketplace, flux RSS, ...).</i>	✓
Requis gd extension <i>Requis pour le traitement des images.</i>	✓
Requis intl extension <i>Requis pour l'internationalisation.</i>	✓
Requis mbstring extension <i>Requis pour la prise en charge des caractères multioctets et la conversion de jeu de caractères.</i>	✓
Requis zlib extension <i>Requis pour la gestion de la communication compressée avec les agents d'inventaire, l'installation de paquets gzip à partir du Marketplace et la génération de PDF.</i>	✓
Requis bcmath extension <i>Requis pour la prise en charge des QR codes.</i>	✓
Requis Libsodium ChaCha20-Poly1305 constante de taille <i>Activer l'utilisation du cryptage ChaCha20-Poly1305 requis par GLPI. Il est fourni par Libsodium à partir de la version 1.0.12.</i>	✓
Requis openssl extension <i>Requis pour l'envoi d'e-mails via SSL/TLS, la gestion des communications chiffrées avec les agents d'inventaire et l'authentification OAuth 2.0.</i>	✓
Requis Permissions pour les fichiers de log	✓
Requis Permissions pour les dossiers de données	✓
Sécurité Version de PHP maintenue <i>Une version de PHP maintenue par la communauté PHP devrait être utilisée pour bénéficier des correctifs de sécurité et de bogues de PHP.</i>	✓
Sécurité Configuration de sécurité pour les sessions <i>Permet de s'assurer que la sécurité relative aux cookies de session est renforcée.</i>	✓
Suggéré exif extension <i>Renforcer la sécurité de la validation des images.</i>	✓
Suggéré ldap extension <i>Active l'utilisation de l'authentification à un serveur LDAP distant.</i>	✓
Suggéré Extensions PHP pour le marketplace <i>Permet le support des formats de paquets les plus communs dans le marketplace.</i>	✓
Suggéré Zend OPcache extension <i>Améliorer les performances du moteur PHP.</i>	✓
Suggéré Extensions émules de PHP <i>Améliorer légèrement les performances.</i>	✓
Suggéré Permissions pour le répertoire du marketplace <i>Active l'installation des plugins à partir du Marketplace.</i>	✓

Il va demander où se trouve le serveur SQL, comme nous l'avons installé sur le même serveur que GLPI on met localhost puis l'utilisateur et le mot de passe mit au-dessus lors de la configuration de la base de données :



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 >

Puis si tous à bien était configuré, vous pourrez sélectionner la base de données créée :



GLPI SETUP

Étape 2

Test de connexion à la base de données



Connexion à la base de données réussie

Veillez sélectionner une base de données :

Créer une nouvelle base ou utiliser une base existante :



db23_glpi

Continuer >

Il va configurer la base de données puis faite continuer>continuer> « utilisé
GLPI »

Vous serrez rediriger sur cette page. L'identifiant et mot de passe par défaut sont
glpi glpi



Connexion à votre compte

Identifiant

Mot de passe

Source de connexion

Base interne GLPI ▼

Se souvenir de moi

Se connecter

GLPI Copyright (C) 2015-2023 Teclib' and contributors

Vous arrivez donc sur cette page et donc à ce point-là on est presque arrivé à la fin :

- Chercher dans le menu
- Parc
- Assistance
- Gestion
- Outils
- Administration
- Configuration

Tableau de bord

Vue personnelle Vue groupe Vue globale Flux RSS Tous

Central

0 Logiciel	0 Ordinateur	0 Matériel réseau	0 Téléphone
0 Licence	0 Moniteur	0 Baie	0 Imprimante

Aucune donnée trouvée	Aucune donnée trouvée	Aucune donnée trouvée
Ordinateurs par Fabricant	Moniteurs par Modèle	Matériels réseau par Statut

5 Utilisateurs	0 Groupe	0 Fournisseur	0 Document
-------------------	-------------	------------------	---------------

Aucune donnée trouvée

Statuts des tickets par mois

0 Ticket	0 Tickets en retard	0 Problème	0 Changement
-------------	------------------------	---------------	-----------------

e) Sécurisation de GLPI

e) Sécurisation de GLPI :

Pour des questions de sécurité nous allons voir comment changé.



- Pour des raisons de sécurité, veuillez changer le mot de passe par défaut pour le(s) utilisateur(s) : glpi post-only tech normal
- Pour des raisons de sécurité, veuillez supprimer le fichier : install/install.php

Aller dans Administration>Utilisateurs :

IDENTIFIANT	NOM DE FAMILLE	COURRIELS	TÉLÉPHONE	LIEU	ACTIF
acheteur					Oui
glpi					Oui
glpi-system	Support				Oui
normal					Oui
post-only					Oui
tech					Oui

Choisissez un compte ou vous souhaité changé le mot de passe :

Identifiant	<input type="text" value="glpi"/>
Nom de famille	<input type="text"/>
Prénom	<input type="text"/>
Mot de passe	<input type="password" value="....."/>
Confirmation mot de passe	<input type="password" value="....."/>

Puis faite sauvegarder.



Vous devriez plus voir le message :



Également, on doit supprimé aussi install.php, cela évitera tous problème si une personne relance une réinstallation. Il se trouve dans /var/www/html/glpi/install.

Faite rm install.php :

```
empty_data.php  index.php  install.php  migrations  mysql  update
root@SRV-GLPI:/var/www/html/glpi/install# rm in
index.php      install.php
root@SRV-GLPI:/var/www/html/glpi/install# rm install.php
```

Dans un second temps, on va sécurisé php-fpm que l'on a activé précédament. La documentations GLPI recommande cela.

On doit sécurisé les cookie, on va donc allé ouvrir le fichier php.ini dans /etc/php/8.4/fpm/ :

```
sudo nano /etc/php/8.4/fpm/php.ini
```

Nous cherchons "**session.cookie_httponly**" pour le mettre sur "**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
```

On viens d'activé la sécurité de base pour les cookies, maintenant nous allons configuré la fasson dont sont géré les cookie pour évité certaines attaques comme CSRF (Cross-Site Resquest Forgery).

Nous cherchons donc maintenant "**session.cookie_samesite**" pour le mettre sur la valeur "**Lax**" :

```
; Add SameSite attribute to cookie to help mitigate Cross-Site Request Forgery (CSRF/XSRF)
; Current valid values are "Strict", "Lax" or "None". When using "None",
; make sure to include the quotes, as `none` is interpreted like `false` in ini files.
; https://tools.ietf.org/html/draft-west-first-party-cookies-07
session.cookie_samesite = Lax
```

Pour finir, on doit édité notre fichier de configuration apache2 pour le forcé apache2 à utilisé php-fpm :

```
GNU nano 8.4 /etc/apache2/sites-enabled/glpi.conf
<VirtualHost *:80>
  ServerName glpi.lerenard.eu
  DocumentRoot /var/www/html/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/html/glpi/public"

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

    RewriteEngine On

    # Ensure authorization headers are passed to PHP.
    # Some Apache configurations may filter them and break usage of API, CalDAV, ...
    RewriteCond %{HTTP:Authorization} ^(.+)$
    RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]

    # 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/php8.4-fpm.sock|fcgi://localhost/"
  </FilesMatch>
</VirtualHost>
```

Puis redémarrons :

```
sudo systemctl restart apache2
```

```
root@GLPI-VM:/home/glpi# sudo systemctl restart apache2
root@GLPI-VM:/home/glpi# |
```

Voilà la fin de la sécurisation de GLPI. Mais cela devrais évolué avec la mise en place d'https dans le futur.