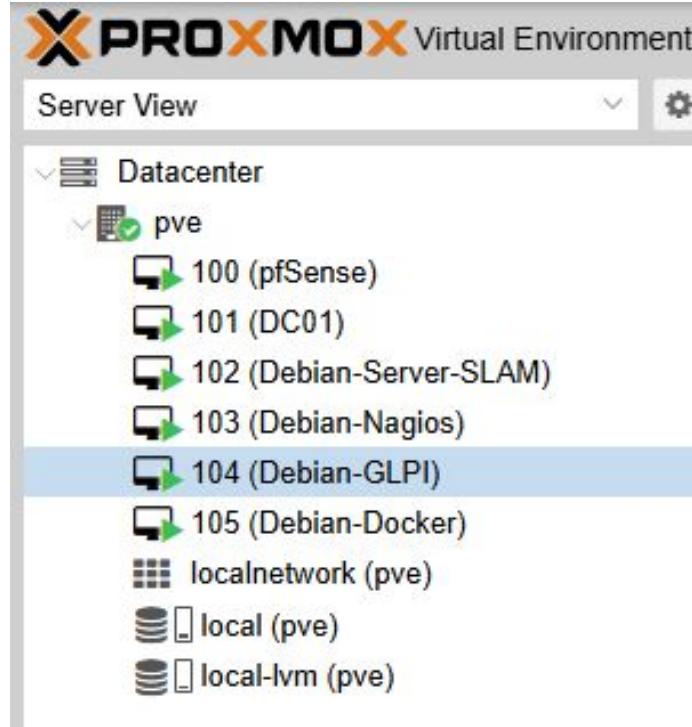


# Installation de GLPI

GLPI

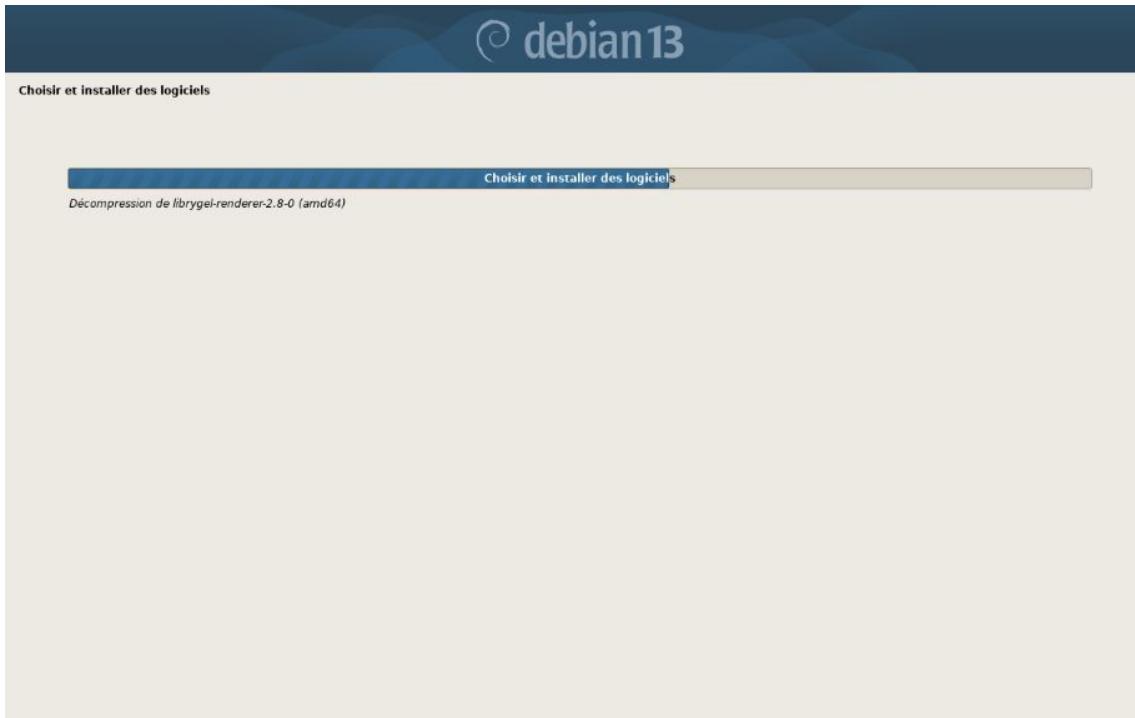
# 1. Téléchargement et intégration Proxmox

Création d'une VM Debian 13 nommée Debian-GLPI avec un ISO ajoutée au pve des vm.



## 2. Lancement et installation

Ouverture puis lancement de la phase d'installation de debian 13 avec interface.



### 3. Téléchargement des composants GLPI

Tout d'abord mise à jour des paquets : `sudo apt-get update && sudo apt-get upgrade`

Pour la suite nous devrons installer Apache2, MariaDB, & PHP :

```
sudo apt-get install apache2 php mariadb-server
```

```
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
```

```
sudo apt-get install php-ldap
```

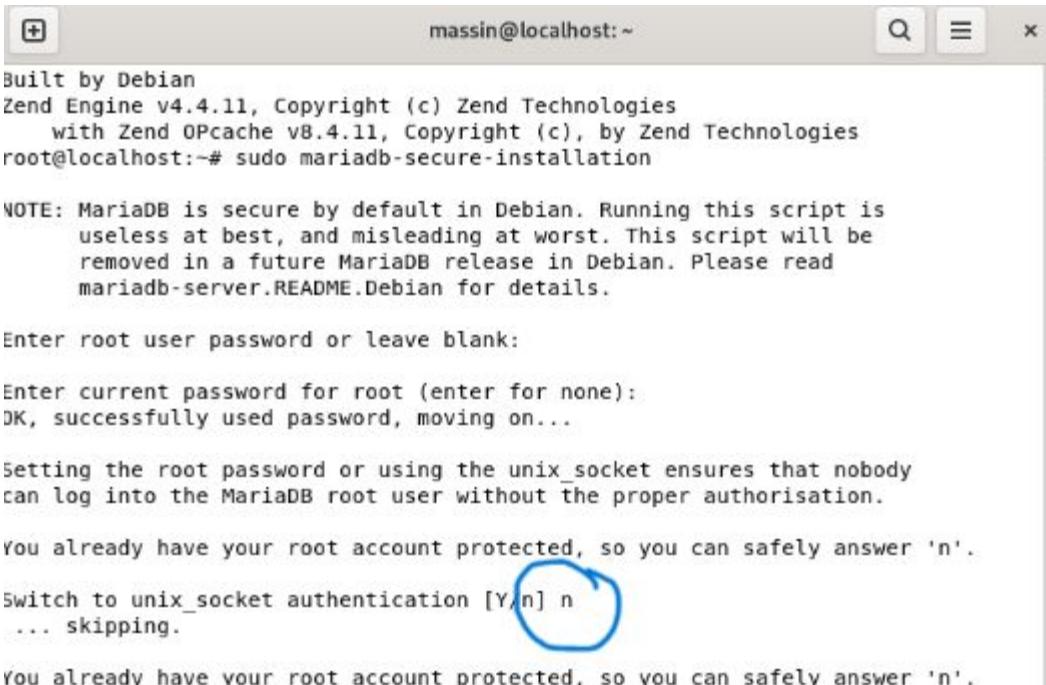
## 4. Préparation de la base de données

Pour préparer la mise en place de cette base de données, nous entrons la commande suivante : `sudo mysql_secure_installation`

(Permet d'ajouter une sécurité à la base de données avant création).

## 4. Préparation de la base de données

A la suite de la commande précédente, MariaDB vous interrogera pour connaître les conditions de sécurité que vous souhaitez appliquer, il faudra appliquer les réponses suivantes :



```
Built by Debian
Zend Engine v4.4.11, Copyright (c) Zend Technologies
    with Zend OPcache v8.4.11, Copyright (c), by Zend Technologies
root@localhost:~# 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'.
```

Répondre oui aux questions suivantes.

# 5. Création de la base de données

Tout d'abord se connecter à MariaDB : `sudo mysql -u root -p`

Entrer ensuite les commandes suivantes :

**CREATE DATABASE db23\_glpi;**

**GRANT ALL PRIVILEGES ON db23\_glpi.\* TO glpi\_adm@localhost IDENTIFIED BY "Menfou-123";**

**FLUSH PRIVILEGES;**

**EXIT**

```
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.11.3-MariaDB-1 Debian 12

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> CREATE DATABASE db23_glpi;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON db23_glpi.* TO glpi_adm@localhost IDENTIFIED BY
Query OK, 0 rows affected (0.003 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> EXIT
Bye
glpi_adm@SRV-GLPI:~$ █
```

## 6. Installation directe GLPI

Pour procéder à l'installation, accéder à cd /tmp puis utiliser la commande :

```
wget https://github.com/glpi-project/glpi/releases/download/10.0.10/glpi-10.0.10.tgz  
root@localhost:~# cd /tmp  
root@localhost:/tmp# wget https://github.com/glpi-project/releases/download/10.0.  
.10/glpi-10.0.10.tgz
```

Puis utiliser la commande suivante pour décompresser l'archive

```
sudo tar -xzvf glpi-10.0.10.tgz -C /var/www/
```

```
root@localhost:/tmp# sudo tar -xzvf glpi-10.0.10.tgz -C /var/www/  
glpi/  
glpi/css_compiled/  
glpi/css_compiled/css_install.min.css  
glpi/css_compiled/css_palettes_aerialgreen.min.css  
glpi/css_compiled/css_palettes_auror.min.css  
glpi/css_compiled/css_palettes_auror_dark.min.css  
glpi/css_compiled/css_palettes_autumn.min.css  
glpi/css_compiled/css_palettes_classic.min.css  
glpi/css_compiled/css_palettes_clockworkorange.min.css  
glpi/css_compiled/css_palettes_dark.min.css  
glpi/css_compiled/css_palettes_darker.min.css  
glpi/css_compiled/css_palettes_flood.min.css  
glpi/css_compiled/css_palettes_greenflat.min.css  
glpi/css_compiled/css_palettes_hipster.min.css  
glpi/css_compiled/css_palettes_icecream.min.css
```

## 7. Configuration du serveur Web

Pour définir Apache2 en tant que propriétaire sur les fichiers GLPI, créer le dossier GLPI et lui attribuer les droits, puis déplacer le fichier de config glpi dans ce dossier, ce qui donne :

```
root@localhost:/tmp# sudo chown www-data /var/www/glpi/ -R
root@localhost:/tmp# sudo mkdir /etc/glpi
root@localhost:/tmp# sudo chown www-data /etc/glpi/
root@localhost:/tmp# sudo mv /var/www/glpi/config /etc/glpi
```

## 8. Idem pour var/lib/glpi

Même chose pour le répertoire /var/lib/glpi :

```
root@localhost:/tmp# sudo mkdir /var/lib/glpi
mkdir: cannot create directory '/var/lib/glpi': Le fichier existe
root@localhost:/tmp# sudo chown www-data /var/lib/glpi
root@localhost:/tmp# sudo mv /var/www/glpi/files /var/lib/glpi
root@localhost:/tmp#
```

---

## 9. Création des logs

Création du répertoire de stockage :

```
sudo mkdir /var/log/glpi
```

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

# 10. Création des fichiers de configuration

Création du fichier de configuration downstream.php :

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

Avec ajout du contenu suivant :

```
<?php  
  
define('GLPI_CONFIG_DIR', '/etc/glpi/');  
  
if (file_exists(GLPI_CONFIG_DIR . '/local_define.php')) {  
  
    require_once GLPI_CONFIG_DIR . '/local_define.php';  
  
}
```

## 10. Création des fichiers de configuration

Suite à ça création du fichier suivant : `sudo nano /etc/glpi/local_define.php`

Puis coller la configuration suivante afin de préparer les chemins files et log :

```
<?php  
  
define('GLPI_VAR_DIR', '/var/lib/glpi/files');  
  
define('GLPI_LOG_DIR', '/var/log/glpi');
```

# 11. Configuration Apache2

Tout d'abord nous allons créer un fichier de configuration à l'aide de la commande suivante :

```
sudo nano /etc/apache2/sites-available/support.it-connect.tech.conf
```

Puis copier la configuration suivante et enregistrer le fichier : <https://pastebin.com/E8a2saMj>

## 12. Activation du site dans Apache2

Nous allons maintenant activer le site it-connect et supprimer le site associé par défaut, à l'aide des deux commandes suivantes :

```
sudo a2ensite support.it-connect.tech.conf
```

```
sudo a2dissite 000-default.conf
```

Maintenant activer le modèle rewrite pour la réécriture : `sudo a2enmod rewrite`

Puis redémarrer Apache2 : `sudo systemctl restart apache2`

## 13. Installation de PHP8.4-FPM

Tout d'abord, installer PHP8.4-FPM (ou la version à jour, voir avec la commande `php -v`) pour le moteur de script : `sudo apt-get install php8.4-fpm`

Ensuite entrer ces trois commandes afin de configurer Apache2, puis le relancer :

```
sudo a2enmod proxy_fcgi setenvif
```

```
sudo a2enconf php8.4-fpm
```

```
sudo systemctl reload apache2
```

## 13. Configuration de PHP8.4-FPM

Pour configurer PHP-FPM, l'ouvrir avec cette commande : `sudo nano /etc/php/8.4/fpm/php.ini`

Dans ce fichier, rechercher "session.cookie\_httponly" et ajouter "on", ce qui donne :

```
; 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
```

Maintenant, il faut redémarrer PHP-FPM afin d'appliquer les modifications :

```
root@localhost:~# sudo systemctl restart php8.4-fpm.service
```

# 14. Mise à jour VirtualHost

Entrer à nouveau la commande

```
sudo nano /etc/apache2/sites-available/support.it-connect.tech.conf
```

Puis ajouter entre “Directory” et “VirtualHost” (Ensuite relancer Apache2 avec

`sudo systemctl restart apache2`) :

```
<FilesMatch \.php$>
```

```
    SetHandler "proxy:unix:/run/php/php8.2-fpm.sock|fcgi://localhost/"
```

```
</FilesMatch>
```

Ce qui donne :

```
GNU nano 7.2
/etc/apache2/sites-available/support.it-connect.tech.conf
VirtualHost *:80>
  ServerName support.it-connect.tech
  DocumentRoot /var/www/glpi/public

  # If you want to place GLPI in a subfolder of your site (e.g. your virtual host is se
  # you can use an Alias directive. If you do this, the DocumentRoot directive MUST NOT
  # 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>
```

## 15. Interface Web

Arrivé sur la page web GLPI (ip a, puis votre adresse IP “ens”), choisissez la langue puis cliquer sur “Installer” :



# 15. Interface Web

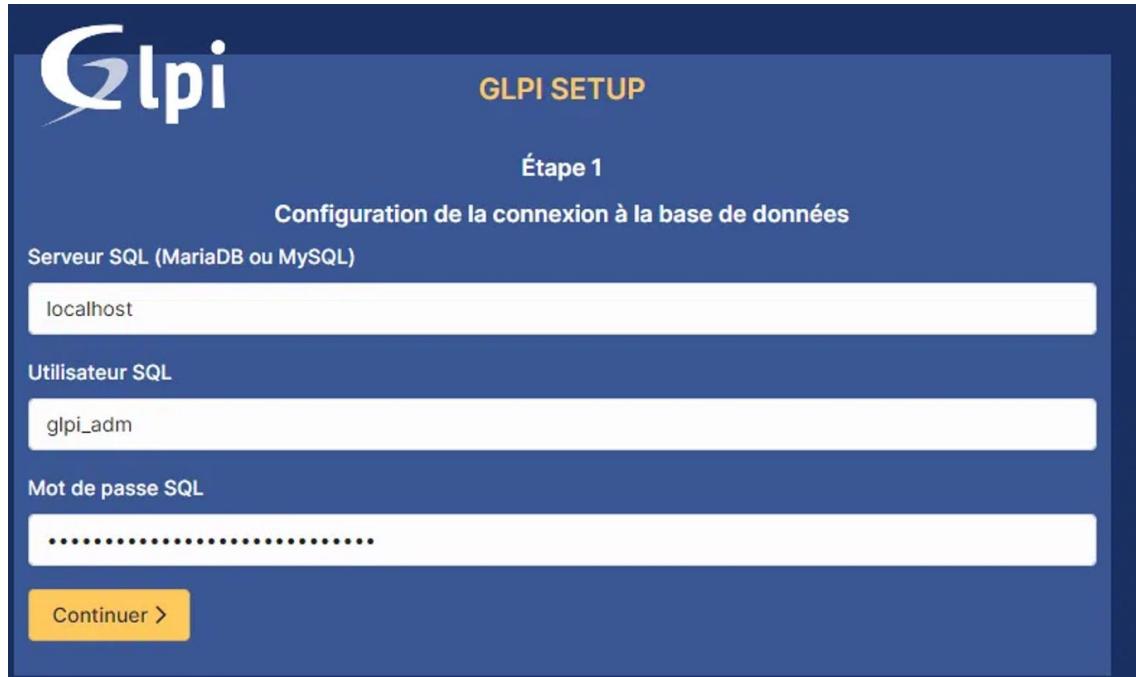
Sur l'écran suivant, tous les prérequis doivent être respectés, ce qui donne :

The screenshot shows the GLPI Setup interface, specifically Step 0: Environment Compatibility Check. The page title is "Vérification de la compatibilité de votre environnement avec l'exécution de GLPI". It lists various system requirements (tests) and their results. Most tests are marked as successful (green checkmark). Some are marked as required (orange box), while others are optional (blue box). A few notes provide additional context for specific requirements.

TESTS EFFECTUÉS	RÉSULTATS
Requis Parser PHP	✓
Requis Configuration des sessions	✓
Requis Mémoire allouée	✓
Requis mysqli extension	✓
Requis Extensions du noyau de PHP	✓
Requis curl extension Requis pour l'accès à distance aux ressources (requêtes des agents d'inventaire, Marketplace, flux RSS, ...).	✓
Requis gd extension Requis pour le traitement des images.	✓
Requis intl extension Requis pour l'internationalisation.	✓
Requis zlib extension 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.	✓
Requis Libsodium ChaCha20-Poly1305 constante de taille Activer l'utilisation du cryptage ChaCha20-Poly1305 requis par GLPI. Il est fourni par libsodium à partir de la version 1.0.12.	✓
Requis Permissions pour les fichiers de log	✓
Requis Permissions pour les dossiers de données	✓
Suggéré Version de PHP supportée Une version officiellement supportée de PHP devrait être utilisée pour bénéficier des correctifs de sécurité et de bogues.	✓
Suggéré Configuration sécurisée du dossier racine du serveur web La configuration du dossier racine du serveur web devrait être '/var/www/glpi/public' pour s'assurer que les fichiers non publics ne peuvent pas être accédés.	✓
Suggéré Configuration de sécurité pour les sessions Permet de s'assurer que la sécurité relative aux cookies de session est renforcée.	✓

## 15. Interface Web

Ensuite, indiquer “localhost” pour le serveur SQL, glpi\_adm pour le username et le mot de passe choisi précédemment :



## 15. Interface Web

Ensuite, il faut choisir la base de données “db23\_glpi” :



# 15. Interface Web

Suivre ensuite les dernières étapes (Importance faible) jusqu'à cet écran de connexion où il faudra se connecter avec l'identifiant "glpi" et le mot de passe "glpi".



Connexion à votre compte

Identifiant

Mot de passe

Source de connexion

Se souvenir de moi

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

## 16. Finalisation

Une fois connecté et l'interface GLPI accessible, l'installation est terminée, par sécurité nous allons exécuter cette dernière commande qui va mettre à jour l'installation :

```
sudo rm /var/www/glpi/install/install.php
```