



## Qu'est-ce que GLPI :

Gestionnaire Libre de Parc Informatique est une application web qui donne la possibilité de visualiser et gérer son système d'information de manière complète : inventaire de toutes les ressources de la société et gestion des tâches administratives et financières.

Ce logiciel est Open Source, ce qui nous permet d'exécuter, de modifier ou de développer le code.

## Les ressources nécessaires :

Pour pouvoir installer et faire fonctionner GLPI nous aurons besoin d'un serveur LAMP.

### Qu'est-ce qu'un serveur LAMP :

Lamp est un acronyme pour Linux, Apache, Maria DB, PHP c'est une pile logiciel qui va permettre de créer et d'héberger un serveur web.

**Apache** est un serveur HTTP, qui va permettre au serveur de communiquer avec le navigateur en utilisant le protocole HTTP(S).

**Maria DB** est un serveur de bases de données. Qui nous servira à stocker, manipuler et à accéder à des listes de données.

**PHP** permet de générer du contenu web dynamiquement.

## Comment l'installer :

### Installation du serveur LAMP :

Pour commencer nous allons mettre à jour notre machine avec la commande suivante :

```
root@GLPI-screen:~# apt update && apt full-upgrade -y
```

### Installation d'Apache2 :

Pour installer le serveur web Apache2 ainsi que sa documentation nous allons taper la commande suivante :

```
root@GLPI-screen:~# apt-get install apache2 apache2-doc
```

Pour continuer nous allons taper y

```
Do you want to continue? [Y/n] y
```

Pour vérifier l'installation nous allons dans un premier temps récupérer l'adresse du serveur web avec la commande suivante :

```
root@GLPI-screen:~# ip a  
inet 192.168.1.118/24
```

Une fois récupéré, nous allons taper sur notre navigateur internet : <http://192.168.1.118>

The screenshot shows a Firefox browser window with the address bar containing '192.168.1.118'. The page itself is the 'Apache2 Debian Default Page'. It features the Debian logo and the title 'Apache2 Debian Default Page'. A red banner at the top right says 'It works!'. The main content explains the purpose of the page and how to replace the index file. Below this is a 'Configuration Overview' section with details about the Debian configuration layout.

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

**Configuration Overview**

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/  
|-- apache2.conf  
|   '-- ports.conf  
|-- mods-enabled  
|   * load
```

Pour qu'Apache démarre automatiquement en même temps que Debian, On saisit la commande suivante :

```
root@GLPI-screen:~# systemctl enable apache2
```

Pour pouvoir vérifier que le service apache fonctionne on utilise la commande suivante :

```
root@GLPI-screen:~# service apache2 status
* apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor>
  Active: active (running) since Tue 2022-11-08 18:00:15 UTC; 4min 4>
    Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 7696 (apache2)
     Tasks: 55 (limit: 115544)
    Memory: 5.9M
      CPU: 71ms
     CGroup: /system.slice/apache2.service
             |-7696 /usr/sbin/apache2 -k start
             |-7697 /usr/sbin/apache2 -k start
             `--7698 /usr/sbin/apache2 -k start
```

## Installation de PHP :

Pour installer PHP nous allons taper la commande suivante :

```
root@GLPI-screen:~# apt-get install php libapache2-mod-php
```

Pour continuer nous allons taper y

```
Do you want to continue? [Y/n] y
```

Nous allons installer toutes les dépendances que GLPI pourrais avoir besoin. Pour sa nous taperons la commande suivante :

```
root@GLPI-screen:~# apt install php-ldap php-imap php-apcu php-xmlrpc ph
p-cas php-mysqli php-mbstring php-curl php-gd php-simplexml php-xml php-
intl php-zip php-bz2 -y
```

```
root@GLPI-screen:~# apt install perl -y
```

## Installation de Maria DB :

Nous allons maintenant installer Maria DB avec la commande suivante :

```
root@GLPI-screen:~# apt-get install mariadb-server
```

Pour continuer nous allons taper y

```
Do you want to continue? [Y/n] y
```

Pour sécuriser l'accès au service de base de données on tapera la commande :

```
root@GLPI-screen:~# mysql_secure_installation
```

On nous demande de taper le mot de passe de l'utilisateur root :

```
Enter current password for root (enter for none): 
```

On entre le nouveau mot de passe :

```
Change the root password? [Y/n] y
New password:
Re-enter new password: □
```

On désactive les connexions anonymes :

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

On désactive la connexion en route depuis une connexion distante :

```
Disallow root login remotely? [Y/n] y□
```

On efface ensuite la base de données de test créée par défaut :

```
Remove test database and access to it? [Y/n] y□
```

On charge les nouveaux paramètres :

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

Nous allons nous connecter aux services de base de données avec le compte root et le mot de passe que nous venons de définir :

```
root@GLPI-screen:~# mysql -u root -p
```

On va maintenant créer la base de données qui nous permettra d'installer GLPI :

```
MariaDB [(none)]> create database db_glpi;
```

Nous allons créer un utilisateur avec tous les priviléges :

```
MariaDB [(none)]> grant all privileges on glpidb.* to glpiuser@localhost
identified by "Ipssi2020";
```

Une fois fini nous allons quitter le service SQL avec la commande `exit`

Nous allons désactiver l'indexation des fichiers de configuration de GLPI afin de sécuriser l'accès au répertoire :

```
root@GLPI-screen:~# nano /etc/apache2/sites-available/000-default.conf□
```

En dessous de la ligne `DocumentRoot` nous allons ajouter les lignes suivantes :

```
<Directory /var/www/html>
Options Indexes FollowSymLinks
AllowOverride All
Require all granted
</Directory>
```

```
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html
<Directory /var/www/html>
    Options Indexes FollowSymLinks
    AllowOverride All
    Require all granted
</Directory>
```

Une fois le fichier modifier et sauvegarder nous allons redémarrer les service Apache2 et Maria DB :

```
root@GLPI-screen:~# service apache2 restart
root@GLPI-screen:~# service mariadb restart
```

## Installation de GLPI :

Pour installer GLPI nous devons nous rendre dans le répertoire temporaire :

```
root@GLPI-screen:~# cd /tmp
```

Puis on installe la dernière version de GLPI :

```
root@GLPI-screen:/tmp# wget https://github.com/glpi-project/glpi/releases/download/9.5.2/glpi-9.5.2.tgz
```

On va ensuite décompresser l'archive GLPI avec la commande suivante :

```
root@GLPI-screen:/tmp# shopt -s dotglob
```

Puis on va supprimer les fichier index.html qu'apache à générer (page d'accueil d'Apache) :

```
root@GLPI-screen:/tmp# rm /var/www/html/index.html
```

Nous allons copier les fichier GLPI dans Apache2 :

```
root@GLPI-screen:/tmp# cp -r glpi/* /var/www/html/
```

Puis nous allons ajouter les droits à Apache2 de ces nouveaux fichiers :

```
root@GLPI-screen:/tmp# chown -R www-data /var/www/html
```

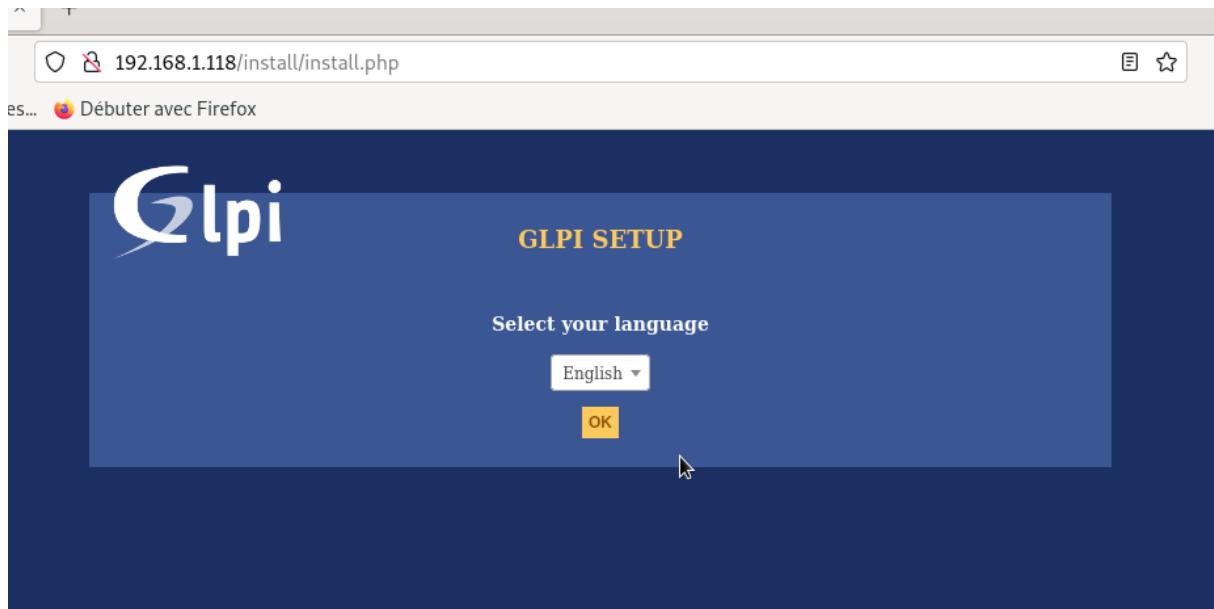
## Accès à GLPI :

Nous pouvons des à présent utiliser GLPI sur notre navigateur :

Pour sa on doit se rendre sur notre navigateur internet et taper l'url suivant :

<http://192.168.1.118/>

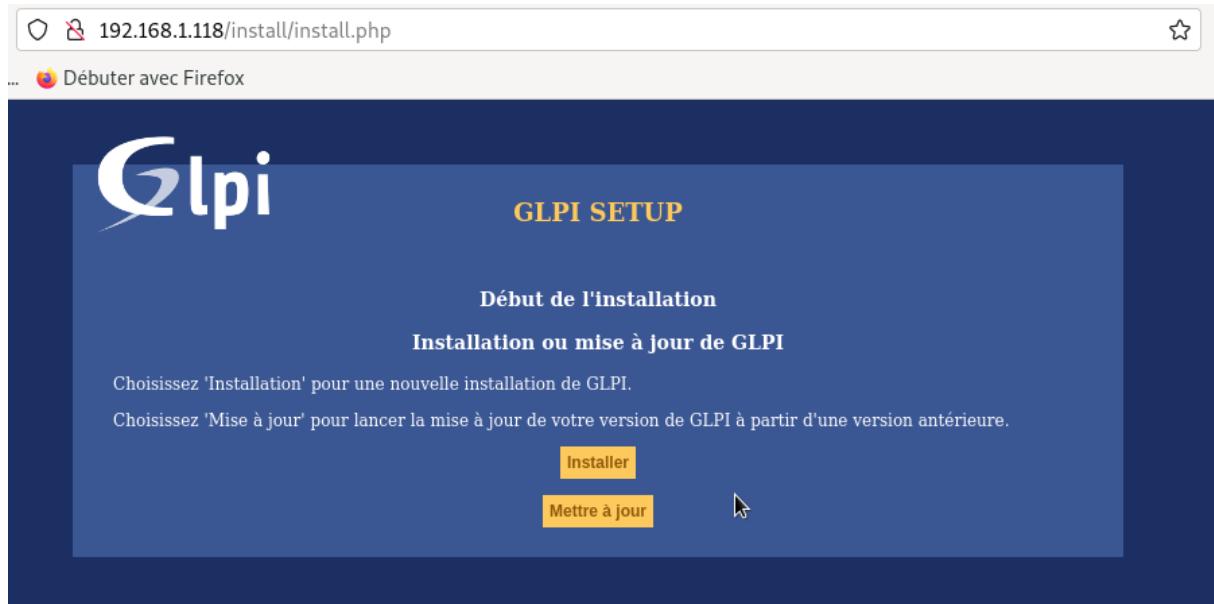
Une fois sur la page nous sélectionnerons comme langue le Français et on clique sur OK :



Puis on accepte les conditions d'utilisation et on clique sur continuer :



On clique sur installer :



Puis sur continuer :

## Étape 0

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

Tests effectués	Résultats
Test du Parseur PHP	✓
Test des sessions	✓
Test de la mémoire allouée	✓
Test de l'extension mysqli	✓
Test de l'extension ctype	✓
Test de l'extension fileinfo	✓
Test de l'extension json	✓
Test de l'extension mbstring	✓
Test de l'extension iconv	✓
Test de l'extension zlib	✓
Test de l'extension curl	✓
Test de l'extension gd	✓
Test de l'extension simplexml	✓
Test de l'extension intl	✓
Test de l'extension ldap	✓
Test de l'extension apcu	✓
Test de l'extension Zend OPcache	✓
Test de l'extension xmlrpc	✓
Test de l'extension CAS	✓
Test de l'extension exif	✓
Test de l'extension zip	✓
Test de l'extension bz2	✓
Test de l'extension sodium	✓
Test d'écriture des fichiers de journal	✓
Test d'écriture du fichier de configuration	✓
Test d'écriture de fichiers documents	✓
Vérification des droits d'écriture du fichier de sauvegarde	✓
Test d'écriture des fichiers de sessions	✓
Test d'écriture des fichiers des actions automatiques	✓
Vérification des droits d'écriture des fichiers graphiques	✓
Test d'écriture des fichiers de verrouillage	✓
Test d'écriture des documents des plugins	✓
Test d'écriture des fichiers temporaires	✓
Test d'écriture des fichiers de cache	✓
Test d'écriture de fichiers RSS	✓
Test d'écriture des fichiers téléchargés	✓
Test d'écriture de fichiers photos	✓
Vérification des permissions d'écriture du dossier marketplace	✓
L'accès web au répertoire des fichiers est protégé	✓

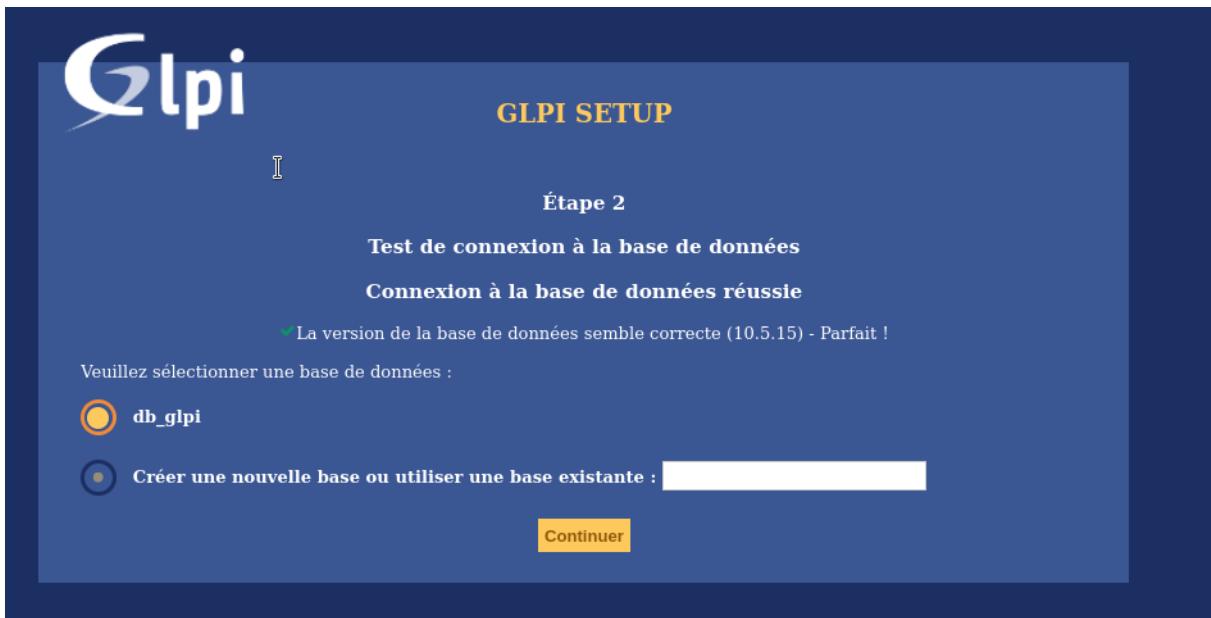
Voulez-vous continuer ?

[Continuer](#) [Réessayer](#)

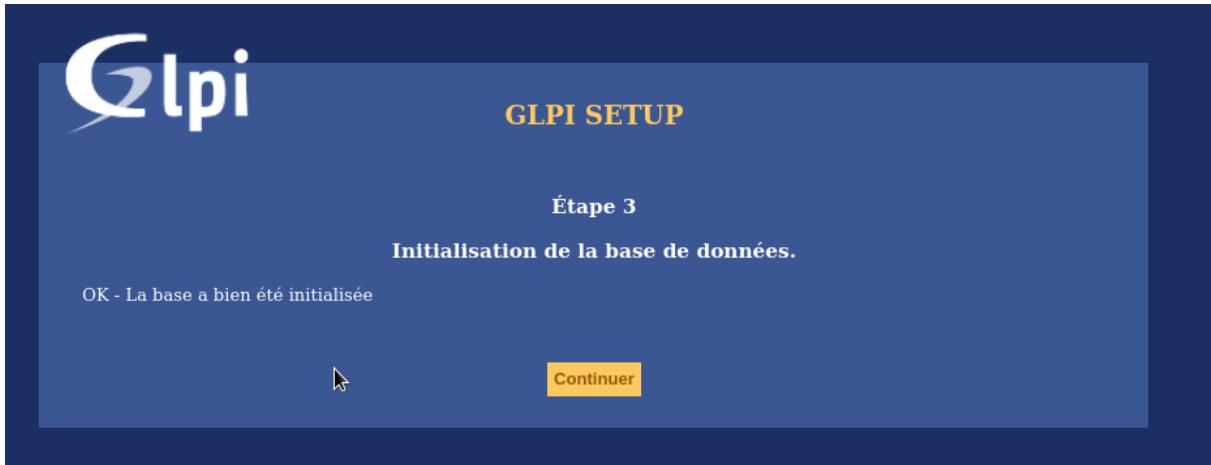
Pour la première ligne nous allons saisir localhost afin de spécifier que la machine héberge le site web de GLPI et la base de données. Ensuite on entre le nom de l'utilisateur et le mot de passe qui aura tous les privilèges.



On sélectionne la base de données que nous avons créés pour GLPI :



On fait continuer :



On fait continuer :



On clique sur utiliser GLPI :



Pour la première connexion l'identifiant est par défaut glpi en nom d'utilisateur et glpi en mot de passe :



Et nous voilà connecter sur GLPI :

The screenshot shows a web browser window with the URL [192.168.1.118/front/central.php](http://192.168.1.118/front/central.php). The page is titled "Interface standard". At the top, there is a navigation bar with tabs: Parc, Assistance, Gestion, Outils, Administration, Configuration, and a search bar. A yellow warning box is prominently displayed, containing the text: "Pour des raisons de sécurité, veuillez changer le mot de passe par défaut pour le(s) utilisateur(s) : gipi post-only tech normal" and "Pour des raisons de sécurité, veuillez supprimer le fichier : install/install.php". Below the warning, there is a section with four status indicators: "Logiciel" (green), "Ordinateur" (red), "Matériel réseau" (light blue), and "Téléphone" (light blue). On the right side of the screen, there is a large, empty rectangular area.

Pour le message d'avertissement il nous indique de changer les mots de passes des 4 utilisateurs par défaut et de supprimer le fichier : `install/install.php`

Pour supprimer le fichier `install/install.php` on tape la commande suivante :

```
rm/var/www/html/install/install.php
```