

Tp installation glpi sur debian 12

mise a jour de debian avec : `apt-get update && sudo apt-get upgrade`

Installation du socle LAMP avec : `apt-get install apache2 php mariadb-server`

Installer toutes les extensions nécessaires au bon fonctionnement de GLPI avec : `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`

`apt-get install php-ldap`

Préparation d'une base de données pour glpi avec : `mysql_secure_installation`

Connexion a mariadb avec : `mysql -u root -p`

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

aller dans le répertoire `cd /tmp` et télécharger glpi : `wget https://github.com/glpi-project/glpi/releases/download/10.0.10/glpi-10.0.10.tgz`

ensuite faire : `tar -xzf glpi-10.0.10.tgz -C /var/www/`

puis : `chown www-data /var/www/glpi/ -R`

```
mkdir /etc/glpi  
chown www-data /etc/glpi/  
mv /var/www/glpi/config /etc/glpi  
mkdir /var/lib/glpi  
chown www-data /var/lib/glpi/  
mv /var/www/glpi/files /var/lib/glpi  
mkdir /var/log/glpi  
chown www-data /var/log/glpi
```

`nano /var/www/glpi/inc/downstream.php` ajouter le contenu suivant le fichier :

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

créer un second fichier : `nano /etc/glpi/local_define.php`
et ajouter le contenu :

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

Préparer la configuration apache2 :

nano /etc/apache2/sites-available/yanis.local.conf :

```
<VirtualHost *:80>
```

```
    ServerName yanis.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>
```

nano /etc/apache2/sites-available/yanis.local.conf

```
/usr/sbin/a2dissite 000-default.conf
```

```
/usr/sbin/a2enmod rewrite
```

```
systemctl restart apache2
```

le module php8.2-fpm ne passe pas il faut donc l'installer en sudo malgré le root :

```
apt-get install sudo
```

```
sudo /usr/sbin/a2enconf php8.2-fpm
```

```
systemctl restart apache2
```

```
sudo /usr/sbin/a2enconf php8.2-fpm
```

```
systemctl reload apache2
```

dans le fichier : nano /etc/php/8.2/fpm/php.ini

faire ctrl+w pour faire une recherche : **session.cookie_httponly** valider la recherche avec entrée puis rajouter 'on' sur la bonne ligne :

```
; 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
```

pour appliquer les changements restart le service :

```
systemctl restart php8.2-fpm.service
```

retourner dans le fichier virtualHost et rajouter les commandes php :

nano /etc/apache2/sites-available/yanis.local.conf

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

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

mettre la commande entre le </Directory> et le </VirtualHost>

enregistrer puis restart apache2 :

```
systemctl restart apache2
```

se connecter avec l'adresse ip (192.168.1.2)

Compétences acquises :

- **Déploiement d'un serveur LAMP** (Linux, Apache, MySQL, PHP) sur un environnement local ou distant.
- **Installation et configuration d'un outil de gestion de parc informatique** (GLPI).
- **Création et gestion d'une base de données MySQL** pour une application métier.
- **Configuration des profils utilisateurs, des groupes, et des droits d'accès** dans GLPI.
- **Mise en place de catégories, règles de traitement et gestion des tickets** pour structurer le support.
- **Gestion et suivi des équipements informatiques** (PC, imprimantes, logiciels, licences, etc.).
- **Amélioration de la traçabilité et du support utilisateurs** via une solution ITSM open source.
- Compréhension du fonctionnement **client/serveur web avec base de données relationnelle**.