**Tutorial : Commandes pour l'installation de GLPI sur Debian 11**

Cette fiche inclut les commandes pour l'installation, les tests, la configuration, et les étapes de sécurisation.

Une image contenant Police, logo, Graphique, Bleu électrique

Description générée automatiquement

**1. Mise à jour du système**

Avant de commencer, assurez-vous que le système est à jour :

sudo apt update && sudo apt upgrade -y

**2. Installation des prérequis**

**2.1 Installer un serveur web (Apache)**

Une image contenant stylos et plumes, outil d’écriture, Police

Description générée automatiquement

sudo apt install apache2 -y

Vérifiez le statut d'Apache :

sudo systemctl status apache2

**2.2 Installer PHP et les extensions nécessaires**

Une image contenant Police, capture d’écran, logo, symbole

Description générée automatiquement

GLPI nécessite PHP avec certaines extensions :

sudo apt install php php-cli php-curl php-mysql php-gd php-xml php-mbstring php-bz2 php-intl php-ldap php-apcu php-zip -y

Vérifiez la version de PHP :

php -v

**2.3 Installer un serveur de base de données (MariaDB)**

Une image contenant Mammifère marin, pinnipède, mammifère, lion de mer

Description générée automatiquement

sudo apt install mariadb-server mariadb-client -y

Assurez-vous que MariaDB est actif :

sudo systemctl status mariadb

**3. Configuration de la base de données**

**3.1 Sécuriser l'installation de MariaDB**

sudo mysql\_secure\_installation

Répondez aux questions pour sécuriser la base de données :

1. Définir un mot de passe pour l’utilisateur root.
2. Supprimer les utilisateurs anonymes.
3. Désactiver la connexion root à distance.
4. Supprimer la base de données test.
5. Recharger les tables de privilèges.

**3.2 Créer une base de données GLPI**

Connectez-vous à MariaDB :

sudo mysql -u root -p

Exécutez les commandes suivantes dans MariaDB :

CREATE DATABASE glpidb CHARACTER SET utf8mb4 COLLATE utf8mb4\_general\_ci;

CREATE USER 'glpiuser'@'localhost' IDENTIFIED BY 'password';

GRANT ALL PRIVILEGES ON glpidb.\* TO 'glpiuser'@'localhost';

FLUSH PRIVILEGES;

EXIT;

**4. Installation de GLPI**

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

**4.1 Télécharger et déployer GLPI**

Téléchargez la dernière version de GLPI :

wget https://github.com/glpi-project/glpi/releases/download/10.0.10/glpi-10.0.10.tgz

Décompressez l’archive :

tar -xvzf glpi-10.0.10.tgz

Déplacez les fichiers dans le répertoire web d’Apache :

sudo mv glpi /var/www/html/

Attribuez les bons droits :

sudo chown -R www-data:www-data /var/www/html/glpi

sudo chmod -R 755 /var/www/html/glpi

**4.2 Configurer Apache**

Créez un fichier de configuration pour GLPI :

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

Ajoutez le contenu suivant :

<VirtualHost \*:80>

ServerAdmin admin@example.com

DocumentRoot /var/www/html/glpi

ServerName glpi.example.com

<Directory /var/www/html/glpi>

Options FollowSymLinks

AllowOverride All

Require all granted

</Directory>

ErrorLog ${APACHE\_LOG\_DIR}/glpi\_error.log

CustomLog ${APACHE\_LOG\_DIR}/glpi\_access.log combined

</VirtualHost>

Activez le site et modifiez la configuration Apache :

sudo a2ensite glpi.conf

sudo a2enmod rewrite

sudo systemctl restart apache2

**5. Tests après l'installation**

**5.1 Accéder à l’interface web**

Ouvrez un navigateur et accédez à l’adresse suivante :

http://<adresse\_IP>/glpi

Suivez les étapes d'installation via l'interface graphique :

1. Sélectionnez la langue.
2. Vérifiez les prérequis système.
3. Configurez la base de données (nom : glpidb, utilisateur : glpiuser, mot de passe : password).

**5.2 Se connecter à GLPI**

Utilisez les identifiants par défaut :

* **Administrateur** : *glpi / glpi*
* **Technicien** : *tech / tech*
* **Utilisateur** : *post-only / post-only ou normal / normal*

**6. Sécurisation post-installation**

**6.1 Supprimer le dossier d'installation**

Une fois GLPI configuré, supprimez le répertoire install/ :

sudo rm -rf /var/www/html/glpi/install

**6.2 Renforcer les permissions**

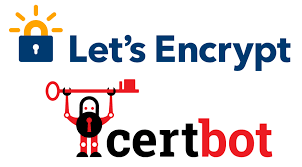
Empêchez les modifications non autorisées :

sudo chmod -R 750 /var/www/html/glpi

sudo chmod -R 770 /var/www/html/glpi/files

sudo chmod -R 770 /var/www/html/glpi/config

**6.3 Configurer un certificat SSL (optionnel à traiter en dernier)**



**Visualisez la vidéo suivante :**

[**Let's encrypt :10 min pour tout comprendre ( Ce que fait Certbot sur votre serveur)**](https://www.youtube.com/watch?v=dokcsaT9Q9k&t=544s)

Installez Certbot pour un certificat SSL gratuit :

sudo apt install certbot python3-certbot-apache -y

sudo certbot --apache

**7. Vérifications et maintenance**



**7.1 Vérifiez le statut des services**

sudo systemctl status apache2

sudo systemctl status mariadb

**7.2 Surveillez les logs pour détecter des problèmes**

Logs Apache :

sudo tail -f /var/log/apache2/error.log

Logs GLPI :

sudo tail -f /var/www/html/glpi/files/\_log/sql-errors.log

**7.3 Sauvegardez régulièrement la base de données**

Créez une sauvegarde :

mysqldump -u root -p glpidb > /backup/glpi\_backup.sql

Restaurer une sauvegarde :

1. Créez une base de données vide sur le serveur cible :

mysql -u root -p -e "CREATE DATABASE glpidb;"

1. Importez le fichier de sauvegarde :

mysql -u root -p glpidb < /backup/glpi\_backup.sql

Planification régulière des sauvegardes :

1. Utilisez un cron job pour automatiser la commande :

crontab -e

1. Ajoutez une tâche quotidienne :

0 2 \* \* \* mysqldump -u root -p[password] glpidb > /backup/glpi\_backup\_$(date +\%F).sql

Compression et sécurisation des sauvegardes :

1. Réduisez la taille des fichiers avec gzip :

mysqldump -u root -p glpidb | gzip > /backup/glpi\_backup.sql.gz

1. Limitez les permissions d’accès au fichier de sauvegarde :

chmod 600 /backup/glpi\_backup.sql

1. Sauvegardez sur un stockage externe ou dans un service cloud sécurisé.

Commenter le script ci-dessous permettant la sauvegarde dans un site distant :

# **definition de l'user de la db MySQL**

DB\_USER="root" #**nom d'user admin pour la connexion à la db MySQL**

DB\_PASSWORD="votre\_mot\_de\_passe" #**mettre nouveau Mdp pour la db MySQL**

DB\_NAME="glpidb" # **nom de la db MySQL à save (glpidb)**

BACKUP\_DIR="/tmp/glpi\_backup" #**folder local où la save sera enregistrée**

REMOTE\_DIR="/backups/glpi" # **folder sur le server distant où la save sera envoyée**

REMOTE\_HOST="192.168.2.11" #**IP du server distant (changer par l’IP correcte)**

REMOTE\_USER="backup\_user" # **nom d'user SSH pour se co au server distant**

DATE=$(date +%Y-%m-%d) #f**ormatage de la date actuelle sous forme annee-mois-jour**

BACKUP\_FILE="$BACKUP\_DIR/glpi\_backup\_$DATE.sql.gz"

#**creer le folder de save si il n'existe pas encore sans dire erreur si il existe**

mkdir -p "$BACKUP\_DIR"

# **effectuer la sauvegarde de la base de données, puis compression avec gzip**

mysqldump -u "$DB\_USER" -p"$DB\_PASSWORD" "$DB\_NAME" | gzip > "$BACKUP\_FILE"

#**vérif du succès de la commande mysqldump (0 = succès)**

if [ $? -eq 0 ]; then

echo "Sauvegarde réussie : $BACKUP\_FILE"

else

echo "Échec de la sauvegarde !" >&2

exit 1

fi

# **transfert du fichier de save vers le server distant via Secure Copy Protocol**

scp "$BACKUP\_FILE" "$REMOTE\_USER@$REMOTE\_HOST:$REMOTE\_DIR"

#**verif du succès de la commande SCP (0 = succes)**

if [ $? -eq 0 ]; then

echo "Transfert réussi vers $REMOTE\_HOST:$REMOTE\_DIR"

else

echo "Échec du transfert !" >&2

exit 1

fi

# **suppression du fichier de save local après le transfert**

rm -f "$BACKUP\_FILE"