

GLPI 11 Installation, configuration et administration

Installation, configuration, sécurisation et maintenance de GLPI 11 sous Debian 12 avec Apache, PHP 8.2 et MariaDB.

- [Accueil / Sommaire](#)
- [Chapitre 1 - Présentation et prérequis](#)
 - [Présentation de GLPI](#)
 - [Architecture et dimensionnement](#)
 - [Prérequis et préparation](#)
- [Chapitre 2 - Installation de GLPI](#)
 - [Installation d'Apache, PHP et MariaDB](#)
 - [Base de données et fuseaux horaires](#)
 - [Téléchargement et organisation des répertoires](#)
 - [Configuration de PHP et Apache](#)
 - [Finalisation de l'installation et comptes par défaut](#)
- [Chapitre 3 - Configuration et Helpdesk](#)
 - [Paramètres généraux](#)
 - [Entités, profils et groupes](#)
 - [Tickets, catégories, SLA et règles](#)
 - [Actions automatiques et cron](#)

- [Notifications et collecteurs de courriels](#)
- [Chapitre 4 - Active Directory, inventaire et plugins](#)
 - [Préparation de LDAP et LDAPS](#)
 - [Utilisateurs, groupes et dépannage LDAP](#)
 - [Installation de GLPI Agent sous Windows](#)
 - [Déploiement de GLPI Agent par GPO](#)
 - [Agent Linux, OAuth et dépannage](#)
 - [Plugins et Marketplace](#)
- [Chapitre 5 - Sécurité et sauvegardes](#)
 - [HTTPS et reverse proxy Caddy](#)
 - [Sécurité, pare-feu et clés de chiffrement](#)
 - [Sauvegarde complète de GLPI](#)
 - [Restauration de GLPI](#)
- [Chapitre 6 - Maintenance et dépannage](#)
 - [Mise à jour de GLPI](#)
 - [Supervision et entretien](#)
 - [Dépannage du serveur Web et de la base](#)
 - [Dépannage de cron, SMTP et de l'inventaire](#)
 - [Checklist de mise en production](#)
 - [Fiche mémo des principales commandes GLPI](#)
 - [Sources officielles](#)

Accueil / Sommaire

GLPI 11 Installation, configuration et administration

Description du livre

Ce livre regroupe les procédures nécessaires pour installer, configurer, sécuriser et maintenir un serveur GLPI 11 sous Debian 12.

Les commandes sont présentées dans leur ordre d'exécution, accompagnées des explications et des contrôles à réaliser.

Public concerné

- Débutants avec GLPI
- Techniciens support
- Administrateurs systèmes et réseaux
- Gestionnaires de parc informatique
- Étudiants en formation informatique

Conventions utilisées

- `<DOMAINE_GLPI>` : nom DNS complet de GLPI, par exemple `glpi.dreamsaphir.fr`.
- `<IP_GLPI>` : adresse IP du serveur GLPI.
- `<VERSION_GLPI>` : version stable à installer.
- `<MOT_DE_PASSE_BDD>` : mot de passe du compte MariaDB GLPI.

- `<SERVEUR_LDAP>` : nom du contrôleur de domaine.
- `<BASE_DN>` : base de recherche LDAP, par exemple `DC=snts,DC=local`.
- `<TAG_AGENT>` : étiquette utilisée pour orienter les inventaires.

Ne tape pas les symboles `<` et `>` dans les commandes : ils indiquent une valeur à remplacer.

Organisation du livre

Chapitre 1 - Présentation et prérequis

- Présentation de GLPI
- Architecture et dimensionnement
- Prérequis et préparation du serveur

Chapitre 2 - Installation de GLPI

- Installation d'Apache, PHP et MariaDB
- Création de la base de données
- Téléchargement et organisation des répertoires
- Configuration de PHP et Apache
- Finalisation de l'installation et comptes par défaut

Chapitre 3 - Configuration et Helpdesk

- Paramètres généraux
- Entités, profils et groupes
- Tickets, catégories, SLA et règles
- Actions automatiques et cron
- Notifications et collecteurs de courriels

Chapitre 4 - Active Directory, inventaire et plugins

- Préparation de LDAP et LDAPS
- Import des utilisateurs et groupes
- Installation de GLPI Agent sous Windows
- Déploiement de l'agent par GPO

- Agent Linux, OAuth et dépannage
- Plugins et Marketplace

Chapitre 5 - Sécurité et sauvegardes

- HTTPS et reverse proxy Caddy
- Pare-feu, permissions et clés de chiffrement
- Sauvegarde complète
- Restauration

Chapitre 6 - Maintenance et dépannage

- Mise à jour de GLPI
- Supervision et entretien
- Dépannage du serveur Web et de la base
- Dépannage de cron, LDAP, SMTP et GLPI Agent
- Checklist de mise en production
- Fiche mémo des principales commandes
- Sources officielles

Chapitre 1 - Présentation et prérequis

Présentation de GLPI, architecture, dimensionnement et préparation.

Présentation de GLPI

Présentation de GLPI

GLPI est une application Web de gestion de parc informatique et de support. Elle permet notamment de gérer :

- les ordinateurs, écrans, imprimantes, équipements réseau et logiciels ;
- les utilisateurs, groupes, profils et entités ;
- les incidents, demandes, problèmes, changements et projets ;
- les contrats, fournisseurs, licences, budgets et documents ;
- l'inventaire automatique avec GLPI Agent ;
- la création de tickets depuis un formulaire ou une boîte de messagerie.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Architecture et dimensionnement

Architecture et dimensionnement

Architecture retenue

Élément	Choix utilisé
Système	Debian 12
Serveur Web	Apache 2
PHP	PHP 8.2
Base de données	MariaDB 10.11 fournie par Debian 12
Code GLPI	<code>/var/www/glpi</code>
Racine Web	<code>/var/www/glpi/public</code>
Configuration	<code>/etc/glpi</code>
Données et documents	<code>/var/lib/glpi/files</code>
Marketplace	<code>/var/lib/glpi/marketplace</code>
Journaux GLPI	<code>/var/log/glpi</code>
Compte du serveur Web	<code>www-data</code>
Nom DNS d'exemple	<code>glpi.dreamsaphir.fr</code>

La racine du site doit impérativement pointer vers le dossier `public` et non vers `/var/www/glpi`. Les fichiers de configuration, les documents et les clés de chiffrement ne doivent pas être directement accessibles depuis Internet.

Ressources indicatives

Environnement	Processeur	Mémoire	Stockage
Laboratoire ou petit parc	2 vCPU	4 Go	30 Go SSD
Petite production	4 vCPU	8 Go	60 Go SSD
Parc important	4 à 8 vCPU	16 Go ou plus	À dimensionner selon les documents et inventaires

Ces valeurs sont des points de départ. La quantité de documents, le nombre d’agents, les plugins, les recherches et le volume de tickets influencent fortement les besoins.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Prérequis et préparation

Prérequis et préparation

Compatibilité GLPI 11

GLPI 11 nécessite au minimum :

- PHP 8.2 ;
- MySQL 8.0 ou MariaDB 10.6 ;
- un serveur Web pris en charge, comme Apache ou Nginx.

Debian 12 fournit PHP 8.2 et MariaDB 10.11 : il convient donc à GLPI 11.

Informations à préparer

Remplacer les valeurs suivantes dans toute la documentation :

Valeur d'exemple	Valeur à renseigner
<code>glpi.dreamsaphir.fr</code>	Nom DNS réel de GLPI
<code>192.168.1.100</code>	Adresse IP du serveur GLPI
<code>11.0.8</code>	Version stable à installer
<code>MOT_DE_PASSE_BDD_TRES_LONG</code>	Mot de passe unique de la base
<code>dc1.snts.local</code>	Contrôleur de domaine réel
<code>DC=snts,DC=local</code>	Base DN de l'annuaire

Ne jamais réutiliser dans la base de données le mot de passe du compte `root`, d'un administrateur Active Directory ou d'un utilisateur GLPI.

DNS

Créer une entrée DNS :

```
glpi.dreamsaphir.fr    A    192.168.1.100
```

Vérification :

```
getent hosts glpi.dreamsaphir.fr
```

Mise à jour du serveur

```
sudo apt update
sudo apt full-upgrade -y
sudo timedatectl set-timezone Europe/Paris
timedatectl
```

Configurer une adresse IP fixe ou une réservation DHCP avant de poursuivre.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Chapitre 2 - Installation de GLPI

Installation du serveur, de la base de données, de PHP et d'Apache.

Installation d'Apache, PHP et MariaDB

3.1 Installer Apache, MariaDB et PHP

```
sudo apt install -y \  
    apache2 mariadb-server cron \  
    libapache2-mod-php8.2 \  
    php8.2 php8.2-cli php8.2-common php8.2-opcache \  
    php8.2-mysql php8.2-curl php8.2-gd php8.2-intl \  
    php8.2-bcmath php8.2-mbstring php8.2-xml \  
    php8.2-zip php8.2-bz2 php8.2-ldap php8.2-apcu \  
    curl wget ca-certificates tar unzip ldap-utils netcat-openbsd
```

Activer et démarrer les services :

```
sudo systemctl enable --now apache2 mariadb  
sudo systemctl status apache2 mariadb --no-pager
```

Contrôler PHP :

```
php -v  
php -m | sort
```

Les modules importants doivent inclure au minimum `bcmath`, `curl`, `dom`, `fileinfo`, `gd`, `intl`, `mbstring`, `mysqli`, `openssl`, `session`, `xml` et `zlib`.

3.2 Sécuriser MariaDB

```
sudo mariadb-secure-installation
```

Recommandations :

- supprimer les utilisateurs anonymes ;
- interdire la connexion distante de `root` ;
- supprimer la base de test ;
- recharger les privilèges.

Sur Debian, le compte MariaDB `root` utilise généralement l'authentification locale par socket. Il n'est pas obligatoire de lui attribuer un mot de passe si l'accès administrateur se fait avec `sudo mariadb`.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Base de données et fuseaux horaires

3.3 Créer la base GLPI

Ouvrir MariaDB :

```
sudo mariadb
```

Exécuter les requêtes suivantes en remplaçant le mot de passe :

```
CREATE DATABASE glpi
  CHARACTER SET utf8mb4
  COLLATE utf8mb4_unicode_ci;

CREATE USER 'glpi'@'localhost'
  IDENTIFIED BY 'MOT_DE_PASSE_BDD_TRES_LONG';

GRANT ALL PRIVILEGES ON glpi.* TO 'glpi'@'localhost';

FLUSH PRIVILEGES;
EXIT;
```

Tester le compte :

```
mariadb -u glpi -p -e "SHOW DATABASES;"
```

Le compte doit voir la base `glpi`.

3.4 Charger les fuseaux horaires dans MariaDB

Cette étape permet à GLPI de proposer correctement les fuseaux horaires :

```
sudo mysql_tzinfo_to_sql /usr/share/zoneinfo | sudo mariadb mysql
sudo systemctl restart mariadb
```

Vérification :

```
sudo mariadb mysql -e \
"SELECT Name FROM time_zone_name WHERE Name='Europe/Paris';"
```

Avec GLPI 11, il ne faut pas accorder au compte applicatif des droits globaux sur les tables système MariaDB.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Téléchargement et organisation des répertoires

3.5 Télécharger GLPI

Toujours télécharger l'archive officielle `glpi-VERSION.tgz`. Les archives GitHub nommées **Source code** ne contiennent pas toutes les dépendances nécessaires et ne doivent pas être utilisées.

```
cd /tmp
GLPI_VERSION="11.0.8"
wget "https://github.com/glpi-project/glpi/releases/download/${GLPI_VERSION}/glpi-${GLPI_VERSION}.tgz"
tar -xzf "glpi-${GLPI_VERSION}.tgz"
sudo mv glpi /var/www/glpi
```

Vérifier la présence des fichiers :

```
ls -la /var/www/glpi
ls -la /var/www/glpi/public
```

3.6 Séparer le code, la configuration et les données

Créer les répertoires persistants :

```
sudo mkdir -p \
  /etc/glpi \
  /var/lib/glpi/files \
  /var/lib/glpi/marketplace \
```

```
/var/log/glpi
```

Copier les données initiales :

```
sudo cp -a /var/www/glpi/config/. /etc/glpi/
sudo cp -a /var/www/glpi/files/. /var/lib/glpi/files/
```

Créer `/var/www/glpi/inc/downstream.php` :

```
sudo tee /var/www/glpi/inc/downstream.php >/dev/null <<'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';
}

PHP
```

Créer `/etc/glpi/local_define.php` :

```
sudo tee /etc/glpi/local_define.php >/dev/null <<'PHP'
<?php
define('GLPI_VAR_DIR', '/var/lib/glpi/files');
define('GLPI_LOG_DIR', '/var/log/glpi');
define('GLPI_MARKETPLACE_DIR', '/var/lib/glpi/marketplace');
PHP
```

Appliquer les permissions :

```
sudo chown -R root:root /var/www/glpi
sudo find /var/www/glpi -type d -exec chmod 0755 {} \;
sudo find /var/www/glpi -type f -exec chmod 0644 {} \;

sudo chown -R www-data:www-data \
    /etc/glpi \
    /var/lib/glpi \
    /var/log/glpi

sudo find /etc/glpi /var/lib/glpi /var/log/glpi \
    -type d -exec chmod 0750 {} \;
sudo find /etc/glpi /var/lib/glpi /var/log/glpi \
```

```
-type f -exec chmod 0640 {} \;
```

Le code de GLPI est en lecture seule pour Apache. Les répertoires contenant la configuration, les documents, le cache, les journaux et la Marketplace restent accessibles en écriture à `www-data`.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Configuration de PHP et Apache

3.7 Configurer PHP

Créer `/etc/php/8.2/apache2/conf.d/99-glpi.ini` :

```
sudo tee /etc/php/8.2/apache2/conf.d/99-glpi.ini >/dev/null <<'INI'
date.timezone = Europe/Paris
memory_limit = 512M
max_execution_time = 600
max_input_vars = 5000
post_max_size = 64M
upload_max_filesize = 64M

session.cookie_httponly = On
session.cookie_samesite = Lax

opcache.enable = 1
opcache.memory_consumption = 128
opcache.interned_strings_buffer = 16
opcache.max_accelerated_files = 10000
opcache.revalidate_freq = 60
INI
```

Lorsque l'instance est accessible uniquement en HTTPS, ajouter :

```
session.cookie_secure = On
```

Redémarrer Apache :

```
sudo systemctl restart apache2
```

Contrôler les valeurs chargées :

```
php -i | grep -E \  
'date.timezone|memory_limit|upload_max_filesize|post_max_size'
```

3.8 Configurer le VirtualHost Apache

Créer `/etc/apache2/sites-available/glpi.conf` :

```
<VirtualHost *:80>  
    ServerName glpi.dreamsaphir.fr  
  
    DocumentRoot /var/www/glpi/public  
  
    <Directory /var/www/glpi/public>  
        Options FollowSymLinks  
        AllowOverride None  
        Require all granted  
  
        RewriteEngine On  
  
        # Transmettre l'en-tête Authorization à GLPI.  
        RewriteCond %{HTTP:Authorization} ^(.+)$  
        RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]  
  
        # Envoyer les requêtes vers le routeur GLPI,  
        # sauf lorsqu'un fichier existe réellement.  
        RewriteCond %{REQUEST_FILENAME} !-f  
        RewriteRule ^(.*)$ index.php [QSA,L]  
    </Directory>  
  
    ErrorLog ${APACHE_LOG_DIR}/glpi-error.log  
    CustomLog ${APACHE_LOG_DIR}/glpi-access.log combined  
</VirtualHost>
```

Activer le site et la réécriture :

```
sudo a2enmod rewrite
sudo a2ensite glpi.conf
sudo a2dissite 000-default.conf
sudo apache2ctl configtest
sudo systemctl reload apache2
```

Le résultat de `apache2ctl configtest` doit être :

Syntax OK

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Finalisation de l'installation et comptes par défaut

3.9 Vérifier les prérequis GLPI

Toujours lancer les commandes de la console GLPI avec le compte du serveur Web afin d'éviter la création de fichiers appartenant à `root` :

```
cd /var/www/glpi  
sudo -u www-data php bin/console glpi:system:check_requirements
```

Corriger toutes les erreurs obligatoires avant de poursuivre.

3.10 Terminer l'installation depuis le navigateur

Ouvrir :

```
http://glpi.dreamsaphir.fr
```

Puis :

1. choisir le français ;
2. accepter la licence ;
3. sélectionner **Installer** ;
4. saisir le serveur MariaDB `localhost` ;
5. saisir l'utilisateur `glpi` et son mot de passe ;
6. sélectionner la base `glpi` ;
7. terminer l'installation.

Installation alternative en ligne de commande

```
cd /var/www/glpi
sudo -u www-data php bin/console db:install \
  --default-language=fr_FR \
  --db-host=localhost \
  --db-name=glpi \
  --db-user=glpi \
  --db-password
```

L'option `--db-password` sans valeur demande le mot de passe de façon interactive et évite de l'enregistrer dans l'historique du shell.

3.11 Comptes créés par défaut

Identifiant	Mot de passe initial	Profil
glpi	glpi	Super-Admin
tech	tech	Technicien
normal	normal	Utilisateur
post-only	postonly	Libre-service

Immédiatement après la première connexion :

1. créer un compte administrateur nominatif ;
2. tester ce nouveau compte ;
3. changer le mot de passe du compte `glpi` et le désactiver si possible ;
4. supprimer ou désactiver les comptes `tech`, `normal` et `post-only` s'ils ne sont pas utilisés ;
5. activer l'authentification multifacteur pour les comptes privilégiés.

Il peut être utile de conserver un compte administrateur local de secours, avec un mot de passe long conservé dans un coffre-fort et utilisé uniquement en cas de panne de l'annuaire.

Chapitre 3 - Configuration et Helpdesk

Paramètres généraux, profils, tickets, SLA et automatisations.

Paramètres généraux

Paramètres généraux

4.1 Paramètres généraux

Dans **Configuration > Générale** :

- vérifier le nom de l'instance ;
- définir l'URL de l'application en HTTPS ;
- sélectionner `Europe/Paris` ;
- vérifier le format des dates ;
- définir l'adresse de courriel de l'administrateur ;
- désactiver les fonctions non utilisées ;
- contrôler la politique de mot de passe ;
- définir la durée de session adaptée à l'organisation.

Entités, profils et groupes

Entités, profils et groupes

4.2 Entités

Une entité sépare les données, les règles et parfois les équipes.

Exemple :

Entité racine

- └ Informatique
- └ Administration
 - └ Agence Rouen
 - └ Agence Amiens

Éviter de multiplier les entités sans besoin réel. Une structure trop complexe augmente le risque d’erreurs de droits, de règles et de visibilité.

4.3 Profils

Profil	Usage conseillé
Super-Admin	Administration technique globale
Admin	Administration fonctionnelle
Technicien	Traitement des tickets et gestion du parc
Observateur	Consultation et suivi
Self-Service	Création et suivi de ses propres demandes

Appliquer le principe du moindre privilège :

- ne pas attribuer Super-Admin aux techniciens ;
 - utiliser des groupes plutôt que des droits individuels ;
 - vérifier la récursivité des droits sur les entités ;
 - tester chaque profil avec un compte non administrateur.
-

4.4 Groupes

Exemples de groupes :

- Support N1 ;
- Support N2 ;
- Réseau ;
- Systèmes ;
- Direction ;
- Utilisateurs Rouen.

Les groupes peuvent servir à l'affectation des tickets, à la visibilité des éléments et à la synchronisation avec Active Directory.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Tickets, catégories, SLA et règles

Tickets, catégories, SLA et règles

4.5 Catégories de tickets

Exemple simple :

Matériel

- └ Poste de travail
- └ Écran
- └ Imprimante

Logiciel

- └ Installation
- └ Dysfonctionnement
- └ Droits d'accès

Réseau

- └ Wi-Fi
- └ VPN
- └ Accès Internet

Compte utilisateur

- └ Création
- └ Mot de passe

Pour chaque catégorie, il est possible de définir :

- un groupe ou technicien responsable ;
- un modèle de ticket ;
- un niveau d'urgence ;
- un SLA ;
- une base de connaissances associée.

4.6 SLA et horaires

Un SLA correspond à un engagement de prise en charge ou de résolution.

Exemple :

Priorité	Prise en charge	Résolution cible
Critique	15 minutes	4 heures
Haute	1 heure	8 heures
Moyenne	4 heures	2 jours ouvrés
Basse	1 jour ouvré	5 jours ouvrés

Configurer d'abord :

1. les calendriers et jours ouvrés ;
2. les périodes de fermeture ;
3. les SLA ;
4. les règles d'affectation.

4.7 Règles

Les règles permettent notamment :

- d'affecter un ticket selon son adresse destinataire ;
- d'orienter un ticket selon sa catégorie ou son demandeur ;
- d'affecter un ordinateur à une entité grâce au tag de l'agent ;
- d'autoriser automatiquement les utilisateurs LDAP ;
- de normaliser les fabricants, modèles et logiciels.

Toujours documenter l'ordre des règles. GLPI les évalue dans un ordre défini et une règle trop générale peut empêcher les suivantes de s'exécuter.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Actions automatiques et cron

Les notifications, collecteurs de courriels, alertes, SLA et opérations de maintenance dépendent des actions automatiques.

5.1 Créer la tâche cron

Créer `/etc/cron.d/glpi` :

```
sudo tee /etc/cron.d/glpi >/dev/null <<'CRON'
* * * * * www-data /usr/bin/php /var/www/glpi/front/cron.php >/dev/null 2>&1
CRON
```

Appliquer les permissions :

```
sudo chown root:root /etc/cron.d/glpi
sudo chmod 0644 /etc/cron.d/glpi
sudo systemctl restart cron
```

Tester manuellement :

```
sudo -u www-data /usr/bin/php /var/www/glpi/front/cron.php
```

5.2 Configurer GLPI

Dans **Configuration > Actions automatiques** :

- utiliser le mode d'exécution **CLI** ;
- vérifier la date de dernière exécution ;
- contrôler les actions `mailgate`, `queuednotification`, `notification`, `closeticket` et `purgeticket` ;

- ne pas régler toutes les actions sur une fréquence d'une minute sans nécessité.

5.3 Vérifier cron

```
systemctl status cron --no-pager  
sudo journalctl -u cron -n 100 --no-pager  
sudo -u www-data /usr/bin/php /var/www/glpi/front/cron.php
```

Si les actions restent bloquées :

```
cd /var/www/glpi  
sudo -u www-data php bin/console task:unlock
```

Cette commande ne doit être utilisée qu'après avoir identifié l'origine du blocage.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Notifications et collecteurs de courriels

6.1 Envoi des notifications

Dans **Configuration > Notifications** :

1. activer les notifications ;
2. saisir le nom et l'adresse de l'expéditeur ;
3. configurer le serveur SMTP ;
4. envoyer un courriel de test ;
5. contrôler la file d'attente des notifications.

Exemple SMTP :

Paramètre	Exemple
Serveur	smtp.example.fr
Port	587
Sécurité	STARTTLS
Authentification	Oui
Expéditeur	glpi@example.fr

Utiliser un compte de service dédié. Lorsque le fournisseur le permet, préférer OAuth 2.0 à un mot de passe permanent.

Test réseau :

```
openssl s_client -starttls smtp -connect smtp.example.fr:587
```

6.2 Collecteur de courriels

Un collecteur lit une boîte et transforme les messages reçus en tickets.

Dans **Configuration > Collecteurs** :

1. ajouter la boîte `support@example.fr` ;
2. configurer IMAP ou OAuth ;
3. tester la connexion ;
4. associer le collecteur à l'entité adéquate ;
5. créer les règles d'affectation ;
6. vérifier que l'action automatique `mailgate` fonctionne en mode CLI.

Points importants :

- utiliser une boîte dédiée, jamais une boîte personnelle ;
- limiter les droits du compte ;
- éviter une boucle entre les notifications GLPI et la boîte collectée ;
- définir la taille maximale des pièces jointes ;
- surveiller les messages refusés ;
- conserver les journaux nécessaires au diagnostic.

Chapitre 4 - Active Directory, inventaire et plugins

LDAP, LDAPS, GLPI Agent, déploiement GPO et Marketplace.

Préparation de LDAP et LDAPS

7.1 Préparer un compte de service

Créer dans Active Directory un compte dédié, par exemple :

```
svc_glpi_ldap
```

Ce compte doit uniquement disposer des droits de lecture nécessaires dans l'annuaire. Il ne doit pas être administrateur du domaine.

7.2 Installer l'autorité de certification AD

Si le certificat LDAPS est signé par une autorité interne, copier son certificat sur le serveur GLPI :

```
sudo cp ad-ca.crt /usr/local/share/ca-certificates/ad-ca.crt
sudo update-ca-certificates
```

Tester LDAPS :

```
openssl s_client \
  -connect dc1.snts.local:636 \
  -servername dc1.snts.local \
  -showcerts
```

Le nom utilisé dans GLPI doit correspondre au certificat du contrôleur de domaine.

7.3 Paramètres d'annuaire GLPI

Dans **Configuration > Authentification > Annuaire LDAP**, ajouter un annuaire :

Champ	Exemple
Nom	Active Directory SNTS
Serveur	ldaps://dc1.snts.local
Port	636
Base DN	DC=snts,DC=local
Root DN	CN=svc_glpi_ldap,OU=Comptes de service,DC=snts,DC=local
Mot de passe	Mot de passe du compte de service
Champ de l'identifiant	sAMAccountName
Champ de synchronisation	objectGUID

Filtre utilisateur possible :

```
(&(objectCategory=person)(objectClass=user)(!(userAccountControl:1.2.840.113556.1.4.803:=2)))
```

Ce filtre sélectionne les utilisateurs et exclut les comptes désactivés.

7.4 Tester depuis Debian

```
ldapsearch -x \  
-H ldaps://dc1.snts.local:636 \  
-D "CN=svc_glpi_ldap,OU=Comptes de service,DC=snts,DC=local" \  
-W \  
-b "DC=snts,DC=local" \  
"(sAMAccountName=alexandre)"
```

Utilisateurs, groupes et dépannage LDAP

7.5 Importer les utilisateurs

Dans **Administration > Utilisateurs** :

1. ouvrir la liaison vers l'annuaire LDAP ;
2. rechercher les nouveaux utilisateurs ;
3. sélectionner les comptes ;
4. les importer ;
5. attribuer le profil et l'entité ;
6. tester la connexion avec un compte standard.

Avant un import massif, configurer les règles d'attribution d'entités et de profils.

7.6 Groupes LDAP

Configurer ensuite l'import et la synchronisation des groupes. Vérifier :

- le champ de nom du groupe ;
 - le champ d'appartenance ;
 - l'utilisation de `memberOf` ;
 - la récursivité des groupes imbriqués ;
 - la correspondance entre groupes AD et profils GLPI.
-

7.7 Erreurs LDAP courantes

A successful bind must be completed

Vérifier :

- que le Root DN est complet ;
- que le mot de passe du compte de service est renseigné ;
- que le compte n'est ni verrouillé ni expiré ;
- que le serveur autorise la méthode de liaison utilisée ;
- que le certificat LDAPS est approuvé par Debian.

Can't contact LDAP server

```
getent hosts dc1.snts.local
nc -vz dc1.snts.local 636
openssl s_client -connect dc1.snts.local:636
```

La connexion fonctionne mais aucun utilisateur n'apparaît

Vérifier :

- la Base DN ;
- le filtre ;
- les droits de lecture du compte de service ;
- le champ `sAMAccountName` ;
- la présence des utilisateurs dans l'OU interrogée.

Conserver au moins un compte administrateur local GLPI afin de pouvoir intervenir lorsque l'Active Directory est indisponible.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Installation de GLPI Agent sous Windows

8.1 Préparer GLPI

Dans GLPI :

1. vérifier que l'inventaire est activé ;
2. définir les règles d'import et de liaison ;
3. prévoir les entités et tags ;
4. vérifier l'URL d'inventaire.

URL courante :

```
https://glpi.dreamsaphir.fr/front/inventory.php
```

Avec le plugin GLPI Inventory installé depuis la Marketplace, certaines tâches utilisent :

```
https://glpi.dreamsaphir.fr/marketplace/glpiinventory/
```

8.2 Windows — installation silencieuse

Télécharger le MSI x64 depuis les publications officielles de GLPI Agent, puis lancer la commande depuis `cmd.exe` en administrateur :

```
msiexec /i GLPI-Agent-1.18-x64.msi /quiet ^  
SERVER="https://glpi.dreamsaphir.fr/front/inventory.php" ^  
RUNNOW=1 ^  
EXECMODE=1 ^
```

TAG="PARC - PRODUCTION"

Options utiles :

Option	Rôle
<code>SERVER</code>	Adresse du serveur d'inventaire
<code>RUNNOW=1</code>	Lance un inventaire après l'installation
<code>EXECMODE=1</code>	Installe l'agent comme service
<code>TAG</code>	Sert aux règles d'affectation
<code>NO_SSL_CHECK=1</code>	Désactive le contrôle TLS — déconseillé

Ne pas utiliser `NO_SSL_CHECK=1` en production. Installer correctement la chaîne de certification sur les postes.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Déploiement de GLPI Agent par GPO

Déploiement de GLPI Agent par GPO

8.3 Déploiement par GPO

Méthode conseillée :

1. placer le MSI sur un partage réseau accessible aux ordinateurs ;
2. créer une GPO liée aux OU contenant les postes ;
3. déployer le MSI par **Configuration ordinateur > Paramètres du logiciel** ou par script de démarrage ;
4. utiliser un tag par site ou population si nécessaire ;
5. tester d'abord sur une OU pilote.

Exemple de script de démarrage :

```
@echo off
set "MSI=\\snts.local\NETLOGON\GLPI-Agent-1.18-x64.msi"
set "SERVER=https://glpi.dreamsaphir.fr/front/inventory.php"

if exist "C:\Program Files\GLPI-Agent\glpi-agent.exe" exit /b 0

msiexec /i "%MSI%" /quiet SERVER="%SERVER%" RUNNOW=1 EXECMODE=1 TAG="ROUEN"
exit /b %errorlevel%
```

8.4 Forcer un inventaire Windows

```
"C:\Program Files\GLPI-Agent\glpi-agent.bat" --force
```

Journal :

```
C:\Program Files\GLPI-Agent\logs\glpi-agent.log
```

Interface locale de l'agent :

```
http://127.0.0.1:62354
```

L'interface intégrée ne doit pas être exposée à tout le réseau sans contrôle.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Agent Linux, OAuth et dépannage

Agent Linux, OAuth et dépannage

8.5 Linux

Après installation du paquet ou de l'ApplImage officiel, créer un fichier spécifique dans `conf.d` plutôt que modifier le fichier principal :

```
sudo tee /etc/glpi-agent/conf.d/server.cfg >/dev/null <<'CFG'
server = https://glpi.dreamsaphir.fr/front/inventory.php
tag = LINUX-PRODUCTION
CFG
```

Forcer un inventaire :

```
sudo glpi-agent --force
```

Contrôler le service et les journaux :

```
sudo systemctl status glpi-agent --no-pager
sudo journalctl -u glpi-agent -n 100 --no-pager
```

8.6 OAuth 2.0 pour l'inventaire

GLPI 11

GLPI 11 permet de créer un client OAuth avec :

- le type d'octroi **Client Credentials** ;
- le scope **inventory** ;
- un identifiant client ;
- un secret client.

Les paramètres correspondants de l'agent sont :

```
OAUTH_CLIENT_ID  
OAUTH_CLIENT_SECRET
```

Cette méthode améliore l'authentification de l'agent, mais le secret doit être protégé lors du déploiement. Valider la compatibilité de la version de l'agent utilisée avant un déploiement général.

8.7 Dépannage de l'agent

Vérifier depuis le poste :

```
Résolution DNS  
Accès HTTPS  
Validité du certificat  
Présence du service GLPI Agent  
URL SERVER  
Journal de l'agent  
Règles d'import GLPI
```

Sous Windows :

```
nslookup glpi.dreamsaphir.fr  
curl.exe -I https://glpi.dreamsaphir.fr  
sc query glpi-agent
```

Une remontée réussie peut malgré tout être rejetée ou placée dans les éléments ignorés si les règles d'import ne permettent pas son association.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Plugins et Marketplace

Plugins et Marketplace

Bonnes pratiques

- installer uniquement les plugins réellement nécessaires ;
- vérifier leur compatibilité avec GLPI 11 avant une mise à jour ;
- faire une sauvegarde avant installation ou mise à jour ;
- tester les plugins sur une instance de préproduction ;
- préférer la Marketplace intégrée ;
- supprimer les plugins abandonnés ;
- ne jamais télécharger un plugin depuis une source inconnue.

Plugins couramment utilisés

Plugin	Fonction
GLPI Inventory	Découverte réseau, SNMP, déploiement et tâches avancées
Fields	Champs supplémentaires
Data Injection	Import de données
Formcreator	Formulaires avancés, surtout pour les environnements encore basés sur GLPI 10
Reports	Rapports supplémentaires

GLPI 11 intègre nativement des formulaires et des types d'actifs personnalisés. Vérifier si un besoin est déjà couvert nativement avant d'ajouter un plugin.

Commandes de gestion

Lister les commandes disponibles :

```
cd /var/www/glpi  
sudo -u www-data php bin/console
```

Installer et activer un plugin :

```
sudo -u www-data php bin/console glpi:plugin:install NOM_DU_PLUGIN  
sudo -u www-data php bin/console glpi:plugin:activate NOM_DU_PLUGIN
```

Désactiver un plugin :

```
sudo -u www-data php bin/console glpi:plugin:deactivate NOM_DU_PLUGIN
```

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Chapitre 5 - Sécurité et sauvegardes

HTTPS, reverse proxy, permissions, sauvegarde et restauration.

HTTPS et reverse proxy Caddy

HTTPS et reverse proxy Caddy

10.1 Exemple avec Caddy

Sur le serveur Caddy :

```
glpi.dreamsaphir.fr {  
    reverse_proxy 192.168.1.100:80  
}
```

Puis :

```
sudo caddy validate --config /etc/caddy/Caddyfile  
sudo systemctl reload caddy
```

Le DNS public doit pointer vers Caddy et les ports 80/443 doivent arriver sur Caddy, pas directement sur le serveur GLPI.

10.2 Conserver l'adresse IP du client

Si Apache n'est accessible que depuis Caddy, activer `mod_remoteip` :

```
sudo a2enmod remoteip
```

Ajouter dans le VirtualHost, en remplaçant l'adresse de Caddy :

```
RemoteIPHeader X-Forwarded-For
RemoteIPTrustedProxy 192.168.1.10
```

Puis :

```
sudo apache2ctl configtest
sudo systemctl reload apache2
```

Ne jamais faire confiance à `X-Forwarded-For` provenant de n'importe quelle adresse.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Sécurité, pare-feu et clés de chiffrement

Sécurité, pare-feu et clés de chiffrement

10.3 Pare-feu

Si Caddy est sur une autre machine, autoriser le port 80 uniquement depuis son adresse :

```
sudo apt install -y nftables
```

La politique exacte dépend du pare-feu déjà présent. Ne pas activer une nouvelle politique restrictive à distance avant d'avoir explicitement autorisé SSH et vérifié la console de secours.

10.4 Mesures indispensables

- utiliser HTTPS uniquement ;
- pointer Apache vers `/var/www/ssl/public` ;
- changer ou désactiver les comptes par défaut ;
- activer la MFA pour les administrateurs ;
- appliquer rapidement les versions de sécurité ;
- sauvegarder la base, les documents, la configuration et les clés ;
- limiter les droits MariaDB à la seule base `glpi` ;
- utiliser LDAPS plutôt que LDAP simple ;
- limiter l'accès SSH aux administrateurs ;
- ne pas exposer MariaDB sur Internet ;
- maintenir Debian, Apache, PHP et les plugins ;

- vérifier régulièrement les journaux ;
 - tester les restaurations ;
 - limiter les données personnelles collectées au strict besoin.
-

10.5 Clés de chiffrement

GLPI utilise des clés pour chiffrer certaines informations sensibles. Les fichiers présents dans `/etc/glpi`, notamment `glpi.key`, `glpicrypt.key`, `oauth.pem` et `oauth.pub` selon la configuration, doivent faire partie de chaque sauvegarde.

La perte de la base seule est grave. La perte de la base accompagnée des clés peut rendre certaines données chiffrées définitivement inutilisables.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Sauvegarde complète de GLPI

Sauvegarde complète de GLPI

Une sauvegarde GLPI complète contient au minimum :

1. la base MariaDB ;
2. `/etc/glpi` ;
3. `/var/lib/glpi/files` ;
4. `/var/lib/glpi/marketplace` ;
5. `/var/www/glpi/inc/downstream.php` ;
6. la configuration Apache et, si utilisé, Caddy.

Un snapshot de VM est pratique avant une opération, mais ne remplace pas une sauvegarde exportée et testée.

11.1 Sauvegarde manuelle

Créer un dossier daté :

```
sudo install -d -m 0700 /var/backups/glpi
BACKUP_DATE="$(date +%F_%H-%M-%S)"
sudo install -d -m 0700 "/var/backups/glpi/${BACKUP_DATE}"
```

Sauvegarder la base :

```
sudo mariadb-dump \
  --single-transaction \
```

```
--quick \  
--routines \  
--triggers \  
glpi \  
| sudo gzip \  
> "/var/backups/glpi/${BACKUP_DATE}/glpi-database.sql.gz"
```

Sauvegarder les fichiers :

```
sudo tar -czf \  
  "/var/backups/glpi/${BACKUP_DATE}/glpi-files.tar.gz" \  
  /etc/glpi \  
  /var/lib/glpi/files \  
  /var/lib/glpi/marketplace \  
  /var/www/glpi/inc/downstream.php \  
  /etc/apache2/sites-available/glpi.conf
```

Contrôler :

```
sudo ls -lh "/var/backups/glpi/${BACKUP_DATE}"  
sudo gzip -t \  
  "/var/backups/glpi/${BACKUP_DATE}/glpi-database.sql.gz"  
sudo tar -tzf \  
  "/var/backups/glpi/${BACKUP_DATE}/glpi-files.tar.gz" \  
  | head
```

Copier ensuite cette sauvegarde vers un autre serveur ou support. Une sauvegarde conservée uniquement sur la VM GLPI ne protège pas contre la perte de cette VM.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Restauration de GLPI

Restauration de GLPI

11.2 Restauration de la base

Activer la maintenance :

```
cd /var/www/glpi  
sudo -u www-data php bin/console glpi:maintenance:enable
```

Recréer une base vide :

```
sudo mariadb
```

```
DROP DATABASE glpi;  
CREATE DATABASE glpi  
    CHARACTER SET utf8mb4  
    COLLATE utf8mb4_unicode_ci;  
GRANT ALL PRIVILEGES ON glpi.* TO 'glpi'@'localhost';  
FLUSH PRIVILEGES;  
EXIT;
```

Restaurer :

```
sudo gzip -dc \  
    /var/backups/glpi/DATE/glpi-database.sql.gz \  
    | sudo mariadb glpi
```

Restaurer les fichiers :

```
sudo tar -xzf \  
    /var/backups/glpi/DATE/glpi-files.tar.gz \  
    -C /var/www/glpi
```

Rétablir les permissions :

```
sudo chown -R www-data:www-data \  
  /etc/glpi \  
  /var/lib/glpi \  
  /var/log/glpi
```

Contrôler puis désactiver la maintenance :

```
cd /var/www/glpi  
sudo -u www-data php bin/console glpi:system:check_requirements  
sudo -u www-data php bin/console db:check_schema_integrity  
sudo -u www-data php bin/console glpi:maintenance:disable
```

Ne jamais restaurer une sauvegarde sur une base ayant déjà subi une migration partielle. Recréer une base vide avant une nouvelle tentative.

Chapitre 6 - Maintenance et dépannage

Mises à jour, supervision, diagnostic et fiche mémo.

Mise à jour de GLPI

Au 27 juillet 2026, la version stable courante est GLPI 11.0.8. Les versions récentes ont inclus plusieurs corrections de sécurité : les mises à jour ne doivent pas être considérées comme purement esthétiques.

12.1 Avant la mise à jour

1. lire les notes de version ;
2. vérifier la compatibilité des plugins ;
3. réaliser une sauvegarde complète ;
4. tester la restauration si la mise à jour est importante ;
5. contrôler le schéma ;
6. activer la maintenance.

```
cd /var/www/glpi
sudo -u www-data php bin/console db:check_schema_integrity
sudo -u www-data php bin/console glpi:maintenance:enable
```

12.2 Télécharger la nouvelle version

```
cd /tmp
NEW_GLPI_VERSION="VERSION_CIBLE"
wget "https://github.com/glpi-project/glpi/releases/download/${NEW_GLPI_VERSION}/glpi-${NEW_GLPI_VERSION}.tgz"
tar -xzf "glpi-${NEW_GLPI_VERSION}.tgz"
```

Remplacer `VERSION_CIBLE` par la version réellement publiée. Ne jamais deviner un numéro de version : le vérifier sur le site officiel.

12.3 Remplacer le code

La documentation officielle déconseille de copier la nouvelle version par-dessus l'ancienne.

Conserver `downstream.php`, puis permuter les dossiers :

```
sudo cp \  
  /var/www/glpi/inc/downstream.php \  
  /tmp/downstream.php  
  
sudo mv /var/www/glpi \  
  "/var/www/glpi-old-$(date +%F_%H-%M-%S)"  
  
sudo mv /tmp/glpi /var/www/glpi  
  
sudo install \  
  -o root -g root -m 0644 \  
  /tmp/downstream.php \  
  /var/www/glpi/inc/downstream.php  
  
sudo chown -R root:root /var/www/glpi  
sudo find /var/www/glpi -type d -exec chmod 0755 {} \  
sudo find /var/www/glpi -type f -exec chmod 0644 {} \;
```

Les données, la configuration et la Marketplace sont conservées dans `/var/lib/glpi` et `/etc/glpi`.

12.4 Migrer la base

```
cd /var/www/glpi  
sudo -u www-data php bin/console glpi:system:check_requirements  
sudo -u www-data php bin/console db:update
```

Pour une mise à jour majeure, vérifier les plugins puis reprendre leur exécution :

```
sudo -u www-data php bin/console plugin:resume_execution
```

Terminer :

```
sudo -u www-data php bin/console db:check_schema_integrity
sudo -u www-data php bin/console glpi:maintenance:disable
sudo apache2ctl configtest
sudo systemctl reload apache2
```

12.5 Contrôles après mise à jour

- connexion administrateur ;
- création d'un ticket ;
- envoi d'une notification ;
- exécution de cron ;
- collecte d'un courriel ;
- connexion LDAP ;
- remontée d'un agent ;
- consultation d'un document ;
- fonctionnement de chaque plugin ;
- absence d'erreur dans les journaux.

Conserver temporairement l'ancien répertoire source et la sauvegarde jusqu'à validation. Le retour arrière nécessite généralement de restaurer **ensemble** l'ancien code, la base et les fichiers correspondant à la même sauvegarde.

Supervision et entretien

Contrôle des services

```
systemctl status apache2 mariadb cron --no-pager
```

Contrôle HTTP

```
curl -I http://127.0.0.1
curl -I https://glpi.dreamsaphir.fr
```

Espace disque

```
df -h
sudo du -sh \
  /var/lib/glpi/files \
  /var/lib/glpi/marketplace \
  /var/log/glpi \
  /var/lib/mysql
```

Base de données

```
sudo mariadb -e \
  "SELECT table_schema AS base_name,
         ROUND(SUM(data_length + index_length) / 1024 / 1024, 2) AS taille_Mo
  FROM information_schema.tables
  WHERE table_schema='glpi'"
```

```
GROUP BY table_schema;"
```

Vérification GLPI

```
cd /var/www/glpi
sudo -u www-data php bin/console glpi:system:check_requirements
sudo -u www-data php bin/console db:check_schema_integrity
```

Journaux

```
sudo tail -n 100 /var/log/apache2/glpi-error.log
sudo tail -n 100 /var/log/glpi/php-errors.log
sudo journalctl -u apache2 -n 100 --no-pager
sudo journalctl -u mariadb -n 100 --no-pager
```

Fréquence d'entretien conseillée

Fréquence	Opérations
Quotidienne	Vérifier les sauvegardes et les erreurs critiques
Hebdomadaire	Vérifier cron, collecteurs, file de notifications et espace disque
Mensuelle	Vérifier les mises à jour GLPI, plugins, Debian et certificats
Trimestrielle	Tester une restauration et revoir les comptes privilégiés
Annuelle	Revoir l'architecture, les SLA, les règles, la conservation et les droits

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Dépannage du serveur Web et de la base

14.1 Page blanche ou erreur 500

```
sudo apache2ctl configtest
sudo tail -n 100 /var/log/apache2/glpi-error.log
sudo tail -n 100 /var/log/glpi/php-errors.log
php -v
php -m | sort
```

Puis :

```
cd /var/www/glpi
sudo -u www-data php bin/console glpi:system:check_requirements
```

14.2 Erreur de permissions

```
sudo chown -R www-data:www-data \
    /etc/glpi \
    /var/lib/glpi \
    /var/log/glpi

sudo find /etc/glpi /var/lib/glpi /var/log/glpi \
    -type d -exec chmod 0750 {} \;

sudo find /etc/glpi /var/lib/glpi /var/log/glpi \
    -type f -exec chmod 0640 {} \;
```

Ne pas appliquer `chmod -R 777`.

14.3 Connexion à la base impossible

```
systemctl status mariadb --no-pager  
mariadb -u glpi -p glpi  
sudo ss -lntp | grep 3306
```

Contrôler le fichier :

```
/etc/glpi/config_db.php
```

14.4 Mauvaise racine Web

Le VirtualHost doit contenir :

```
DocumentRoot /var/www/glpi/public
```

et non :

```
DocumentRoot /var/www/glpi
```

14.5 L'URL renvoie la page Apache par défaut

```
sudo a2disssite 000-default.conf  
sudo a2ensite glpi.conf  
sudo apache2ctl configtest  
sudo systemctl reload apache2  
apache2ctl -S
```


Dépannage de cron, SMTP et de l'inventaire

14.6 Les actions automatiques ne se lancent pas

```
systemctl status cron --no-pager  
sudo -u www-data php /var/www/glpi/front/cron.php  
sudo journalctl -u cron -n 100 --no-pager
```

Vérifier également que les actions sont en mode **CLI** dans GLPI.

14.7 Les courriels ne partent pas

Vérifier :

- l'activation des notifications ;
- le serveur et le port SMTP ;
- STARTTLS ou TLS ;
- l'authentification ;
- l'action automatique `queuednotification` ;
- la file d'attente ;
- les règles du pare-feu ;
- l'heure du serveur.

```
openssl s_client -starttls smtp -connect smtp.example.fr:587
```

14.8 Les dates ou fuseaux horaires sont incorrects

```
timedatectl  
php -i | grep date.timezone  
sudo mariadb mysql -e \  
"SELECT Name FROM time_zone_name WHERE Name='Europe/Paris';"
```

14.9 L'inventaire arrive mais le poste n'apparaît pas

Contrôler :

- les éléments ignorés ou refusés ;
- les règles d'import et de liaison ;
- le tag ;
- l'entité ;
- les doublons de numéro de série ou UUID ;
- le journal GLPI Agent ;
- l'URL d'inventaire.

14.10 Diagnostic complet à fournir

En cas de demande d'assistance, collecter :

```
cat /etc/os-release  
php -v  
mariadb --version  
apache2 -v  
apache2ctl -S  
systemctl status apache2 mariadb cron --no-pager  
cd /var/www/glpi
```

```
sudo -u www-data php bin/console glpi:system:check_requirements
sudo tail -n 100 /var/log/apache2/glpi-error.log
sudo tail -n 100 /var/log/glpi/php-errors.log
```

Supprimer les mots de passe, secrets OAuth, cookies, jetons API et données personnelles avant de partager les résultats.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Checklist de mise en production

Installation

- ☐ Debian et les paquets sont à jour.
- ☐ Le serveur utilise une adresse IP stable.
- ☐ Le DNS résout correctement le nom GLPI.
- ☐ Apache pointe vers `/var/www/glpi/public`.
- ☐ PHP respecte les prérequis GLPI 11.
- ☐ MariaDB est sécurisé.
- ☐ Le compte MariaDB GLPI est limité à la base `glpi`.
- ☐ Les fuseaux horaires MariaDB sont chargés.
- ☐ La commande de vérification GLPI ne signale aucune erreur obligatoire.

Sécurité

- ☐ HTTPS est obligatoire.
- ☐ Les comptes par défaut ont été changés ou désactivés.
- ☐ Un administrateur nominatif a été créé.
- ☐ Un compte local de secours est protégé dans un coffre-fort.
- ☐ La MFA est activée pour les comptes privilégiés.
- ☐ Les clés GLPI sont incluses dans les sauvegardes.
- ☐ MariaDB n'est pas exposé à Internet.
- ☐ Le serveur est filtré par pare-feu.
- ☐ Les plugins proviennent de sources fiables.

Exploitation

- ☐ Cron fonctionne en mode CLI.
- ☐ L'envoi d'un courriel de test fonctionne.
- ☐ Le collecteur crée correctement un ticket.
- ☐ LDAP ou LDAPS a été testé avec un utilisateur standard.
- ☐ Les profils et entités ont été contrôlés.
- ☐ Un inventaire GLPI Agent a été reçu.
- ☐ Les règles d'import ont été testées.
- ☐ Les journaux sont surveillés.

Sauvegardes

- ☐ La base est sauvegardée.
- ☐ `/etc/glpi` est sauvegardé.
- ☐ `/var/lib/glpi/files` est sauvegardé.
- ☐ La Marketplace et les plugins sont sauvegardés.
- ☐ Une copie existe hors du serveur GLPI.
- ☐ Une restauration a réellement été testée.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Fiche mémo des principales commandes GLPI

Vérification générale

```
systemctl status apache2 mariadb cron --no-pager  
cd /var/www/glpi  
sudo -u www-data php bin/console glpi:system:check_requirements  
sudo -u www-data php bin/console db:check_schema_integrity
```

Ces commandes contrôlent les services, les prérequis GLPI et l'intégrité du schéma de base de données.

Apache

```
sudo apache2ctl configtest  
apache2ctl -S  
sudo systemctl reload apache2  
sudo tail -n 100 /var/log/apache2/glpi-error.log
```

Elles vérifient la configuration du VirtualHost, affichent les sites actifs et consultent les dernières erreurs.

MariaDB

```
systemctl status mariadb --no-pager  
mariadb -u glpi -p glpi  
sudo mariadb-dump --single-transaction --quick glpi > glpi.sql
```

Ces commandes testent la base et produisent une sauvegarde SQL simple.

Actions automatiques

```
sudo -u www-data php /var/www/glpi/front/cron.php  
sudo journalctl -u cron -n 100 --no-pager
```

Elles forcent les actions automatiques puis affichent les journaux cron.

Maintenance

```
cd /var/www/glpi  
sudo -u www-data php bin/console glpi:maintenance:enable  
sudo -u www-data php bin/console db:update  
sudo -u www-data php bin/console glpi:maintenance:disable
```

À utiliser lors d'une mise à jour, uniquement après une sauvegarde complète.

LDAP et LDAPS

```
getent hosts dc1.snts.local  
nc -vz dc1.snts.local 636  
openssl s_client -connect dc1.snts.local:636 -showcerts
```

Ces commandes vérifient le DNS, le port LDAPS et le certificat du contrôleur de domaine.

GLPI Agent

```
"C:\Program Files\GLPI-Agent\glpi-agent.bat" --force
```

```
sudo glpi-agent --force  
sudo journalctl -u glpi-agent -n 100 --no-pager
```

Ces commandes forcent un inventaire sous Windows ou Linux.

Journaux principaux

```
/var/log/apache2/glpi-error.log  
/var/log/apache2/glpi-access.log  
/var/log/glpi/php-errors.log  
C:\Program Files\GLPI-Agent\logs\glpi-agent.log
```

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.

Sources officielles

- [Téléchargement de la dernière version de GLPI](#)
- [Publications officielles GLPI sur GitHub](#)
- [Prérequis GLPI](#)
- [Installation GLPI](#)
- [Assistant d'installation et comptes par défaut](#)
- [Commandes de la console GLPI](#)
- [Procédure officielle de mise à jour](#)
- [Configuration des fuseaux horaires](#)
- [Documentation générale GLPI](#)
- [Configuration LDAP](#)
- [Actions automatiques](#)
- [Documentation GLPI Agent](#)
- [Publications officielles GLPI Agent](#)
- [Paramètres de l'installateur Windows GLPI Agent](#)

“ **Rappel important** : avant toute modification importante, mise à jour ou ajout de plugin, sauvegarder la base, les documents, la configuration et les clés de chiffrement. Une sauvegarde non restaurée au moins une fois reste une sauvegarde non vérifiée.

“ Documentation GLPI 11 — Debian 12, Apache, PHP 8.2 et MariaDB.