PROCÉDURE TECHNIQUE · BTS SIO SISR

Installer GLPI 11.0.12 sur Ubuntu Server 26.04 (Apache, MariaDB, PHP 8.5)

Déployer une instance GLPI fonctionnelle et durcie : base dédiée, VirtualHost limité à /public, assistant web, sécurisation et sauvegarde.

Auteur	Plot 1
Contexte	TP réalisé en équipe — Plot 1, domaine ngrange.local
Version	2.0 — novembre 2026
Environnement	Ubuntu Server 26.04 LTS · Apache 2.4 · MariaDB 11.8 · PHP 8.5 · GLPI 11.0.12
Durée estimée	45 à 60 min

Identifiants d'accès — Plot 1

Machine / service	Adresse	Identifiant	Mot de passe
SRV-GLPI (SSH)	192.168.21.3	`sio2`	`Azerty13.`
Base MariaDB	localhost / glpi	`glpi`	`Azerty13.`
GLPI (web)	http://192.168.21.3/	`glpi`	`glpi (par défaut, à remplacer)`
Annuaire LDAP	SRV-AD1EX 192.168.21.1	`NGRANGE\administrateur`	`Azerty13.`

Sommaire

1. Objectif
2. Architecture et prérequis
3. Installation et configuration
4. Vérification
5. Dépannage
6. Sécurité et bonnes pratiques
7. Aide-mémoire : utiliser GLPI au quotidien

1. Objectif

GLPI est un outil libre de gestion de parc informatique et de helpdesk (inventaire, tickets, licences). L'objectif est de l'installer sur une machine virtuelle Ubuntu **SRV-GLPI** avec une pile LAMP standard, en appliquant dès le départ le principe de moindre privilège :

- l'application est placée dans `/var/www/glpi` et Apache n'expose que le sous-dossier `/public` ;
- GLPI utilise une base MariaDB `glpi` et un compte SQL dédié `glpi@localhost`, limité à cette base ;
- le compte `root` de MariaDB n'est jamais saisi dans l'assistant web.

2. Architecture et prérequis

Élément	Valeur
Machine	VM **SRV-GLPI** (VMID 203, SRV-GLPI.ngrange.local), Ubuntu Server 26.04 LTS, accès Internet et compte avec `sudo`
Adresse IP	`192.168.21.3` (LAN 192.168.21.0/24, passerelle 192.168.21.254, DNS 192.168.21.1 puis 192.168.21.2) ; vérifier avec `hostname -I`
Serveur web	Apache 2.4, module `mod_rewrite`, module PHP `libapache2-mod-php`
Base de données	MariaDB 11.8 (dépôt Ubuntu) — base `glpi` en utf8mb4, utilisateur `glpi@localhost`
PHP	8.5 (dépôt Ubuntu)
Application	Archive officielle `glpi-11.0.12.tgz`, installée dans `/var/www/glpi`
Poste d'administration	Navigateur sur le même réseau que la VM

> **Captures**
> Les captures proviennent du TP réalisé en équipe : les adresses et noms visibles peuvent différer ; les valeurs à utiliser sont celles du texte (Plot 1).

Compatibilité des versions

Composant	Exigence GLPI 11	Version utilisée	Conclusion
PHP	≥ 8.2	8.5	Contrainte satisfaite (voir réserve)
MariaDB	≥ 10.6	11.8	Contrainte satisfaite
Apache	2.x	2.4	Supporté
GLPI	Dernier correctif de la branche 11.0	11.0.12 (correctif de sécurité du 28/10/2026)	À jour

> **Extensions PHP**
> Obligatoires : curl, gd, intl, mysqli, zlib, bcmath, mbstring, openssl, ainsi que les modules XML et session. Recommandées selon l'usage : zip, bz2, exif, ldap, opcache, sodium. Le contrôle qui fait foi est la commande `glpi:system:check_requirements` (étape 8).

> **Réserve sur PHP 8.5**
> Le `composer.json` de GLPI 11 impose PHP ≥ 8.2 sans borne haute, mais la documentation ne présente pas PHP 8.5 comme une version certifiée à part entière. Le contrôle des prérequis et l'accès au tableau de bord valident cette installation précise ; tester les plugins avant toute mise en production.

3. Installation et configuration

1 Mettre le système à jour

Partir d'un système à jour garantit de récupérer les derniers correctifs de sécurité des paquets avant d'installer la pile. Le redémarrage charge le nouveau noyau éventuel.

```
sudo apt update
sudo apt full-upgrade -y
sudo reboot
```

2 Installer Apache, MariaDB, PHP et ses extensions

Les paquets génériques `php` et `libapache2-mod-php` installent la branche PHP fournie par la distribution, ici 8.5. La seconde commande regroupe les extensions requises ou recommandées par GLPI (`php-xml` apporte aussi dom et simplexml ; `php-mysql` apporte mysqli).

```
sudo apt install -y apache2 mariadb-server wget tar
sudo apt install -y php php-cli libapache2-mod-php \
  php-curl php-gd php-intl php-mysql php-xml php-bcmath \
  php-mbstring php-zip php-bz2 php-ldap php-opcache
sudo systemctl enable --now apache2 mariadb
```

Contrôler le socle installé :

```
php -v
apache2 -v
mariadb --version
systemctl --no-pager --full status apache2 mariadb
```

Attendu : PHP 8.5.x, MariaDB au moins 10.6 (ici 11.8), les deux services `active (running)`.

3 Sécuriser MariaDB

Une installation MariaDB neuve contient des réglages de confort (comptes anonymes, base de test) à retirer avant d'y héberger des données.

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

- Supprimer les utilisateurs anonymes.
- Interdire la connexion root à distance.
- Supprimer la base de test et recharger les privilèges.
- Conserver l'authentification locale (socket) proposée par Ubuntu pour `root`.

4 Créer la base glpi et le compte SQL dédié

La base est préparée avant l'assistant, avec un compte limité à `glpi.*` : en cas de compromission de l'application, les autres bases restent inaccessibles. Le jeu de caractères utf8mb4 évite les problèmes d'accents et d'émojis.

```
sudo mariadb

CREATE DATABASE glpi
  CHARACTER SET utf8mb4
  COLLATE utf8mb4_unicode_ci;
CREATE USER 'glpi'@'localhost'
  IDENTIFIED BY 'Azerty13.';
GRANT ALL PRIVILEGES ON glpi.* TO 'glpi'@'localhost';
FLUSH PRIVILEGES;
SHOW GRANTS FOR 'glpi'@'localhost';
EXIT;
```

> **Mot de passe**
> `Azerty13.` est le mot de passe commun du laboratoire Plot 1. En production : un secret aléatoire et unique, rangé dans un gestionnaire de mots de passe.

Contrôle de connexion avec le compte applicatif (quitter ensuite avec `EXIT;`) : il confirme que l'assistant web pourra joindre la base avec les mêmes paramètres.

```
mariadb -u glpi -p -h localhost glpi
```

> **Fuseaux horaires (facultatif)**
> Pour activer la gestion des fuseaux horaires dans GLPI : `mariadb-tzinfo-to-sql /usr/share/zoneinfo | sudo mariadb mysql`, puis `GRANT SELECT ON mysql.time_zone_name TO 'glpi'@'localhost';` (ajout par rapport au TP initial).

5 Télécharger et déployer GLPI 11.0.12

Utiliser l'archive binaire `glpi-11.0.12.tgz` de la page Releases (elle contient les dépendances déjà compilées), et non l'archive générique « Source code ».

```
cd /tmp
wget -O glpi-11.0.12.tgz \
  https://github.com/glpi-project/glpi/releases/download/11.0.12/glpi-11.0.12.tgz
sudo tar -xzf glpi-11.0.12.tgz -C /var/www/
test -f /var/www/glpi/public/index.php && echo "Archive GLPI extraite"
```

> **Pourquoi /public ?**
> La racine de GLPI contient des dossiers et fichiers sensibles qui ne doivent jamais être servis directement. Apache ne publiera que `/var/www/glpi/public`, où se trouve le routeur web.

6 Appliquer les droits sur les fichiers

Le code reste la propriété de root (Apache le lit mais ne peut pas le modifier) ; seuls les dossiers où GLPI doit écrire sont confiés à `www-data`.

```
sudo chown -R root:www-data /var/www/glpi
sudo find /var/www/glpi -type d -exec chmod 755 {} +
sudo find /var/www/glpi -type f -exec chmod 644 {} +
sudo chown -R www-data:www-data \
  /var/www/glpi/config /var/www/glpi/files /var/www/glpi/marketplace
ls -ld /var/www/glpi/{config,files,marketplace}
sudo -u www-data php /var/www/glpi/bin/console --version
```

Zone	Propriétaire	Rôle
Code GLPI	root:www-data	Lecture par Apache, modification réservée à l'administrateur
config	www-data:www-data	Écriture du fichier de connexion à la base pendant l'installation
files	www-data:www-data	Cache, documents, sessions, données applicatives
marketplace	www-data:www-data	Installation et mise à jour de plugins

La dernière commande doit afficher GLPI 11.0.12.

7 Configurer le VirtualHost Apache

Ce site dédié fait pointer Apache sur `/public` et redirige toutes les URL qui ne correspondent pas à un fichier réel vers `index.php` (réécriture), tout en transmettant l'en-tête `Authorization` nécessaire à l'API de GLPI.

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

```
<VirtualHost *:80>
    ServerName SRV-GLPI.ngrange.local
    DocumentRoot /var/www/glpi/public
    <Directory /var/www/glpi/public>
        Require all granted
        AllowOverride None
        RewriteEngine On
        RewriteCond %{HTTP:Authorization} ^(.+)$
        RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]
        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 module et le site, désactiver le site par défaut, valider la syntaxe puis redémarrer :

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

> **Résultat attendu**
>
> `apache2ctl configtest` affiche « Syntax OK ». Le nom `SRV-GLPI.ngrange.local` n'est utilisable depuis le poste d'administration que si un DNS interne (ou un fichier hosts) le résout vers l'IP de la VM ; sinon, utiliser l'adresse IP.

8 Contrôler PHP et les prérequis GLPI

Ces contrôles évitent de découvrir une extension manquante au milieu de l'assistant.

```
php -m | grep -E
'bcmath|curl|dom|fileinfo|gd|intl|mbstring|mysqli|openssl|session|xml|zip|zlib'
sudo -u www-data php /var/www/glpi/bin/console glpi:system:check_requirements
hostname -I
curl -I http://127.0.0.1/
```

Poursuivre uniquement si aucun prérequis obligatoire n'est signalé manquant (les recommandations facultatives peuvent attendre). `curl -I` doit renvoyer une réponse HTTP (200 ou une redirection 302 vers l'assistant).

9 Dérouler l'assistant web (langue, licence, base)

Depuis le poste d'administration, ouvrir `http://192.168.21.3/` ou `http://SRV-GLPI.ngrange.local/`. Les écrans peuvent légèrement varier selon la version, les choix restent les mêmes.

Écran	Action
Langue	Français, puis valider
Licence	Lire et accepter la licence GNU GPL
Type d'opération	Installer (installation neuve)
Vérification de l'environnement	Tous les prérequis obligatoires doivent être au vert
Étape 1 – connexion SQL	Serveur `localhost`, utilisateur `glpi`, mot de passe défini à l'étape 4
Étape 2 – choix de la base	Sélectionner la base existante `glpi` (ne pas en créer une nouvelle)

> **Ne jamais utiliser root**
>
> En cas d'échec de connexion, vérifier l'hôte, l'utilisateur et le mot de passe du compte `glpi` plutôt que de basculer sur root.

10 Initialiser la base et terminer

L'assistant crée la structure de tables et les données par défaut. Laisser la progression atteindre 100 % sans fermer l'onglet.



Figure 1 – Étape 3 de l'assistant : initialisation de la base (81 %), structure créée et données par défaut importées

Choisir ensuite la préférence de télémétrie (facultative, sans effet sur le fonctionnement), terminer l'assistant puis se connecter avec le compte administrateur par défaut indiqué à l'écran.

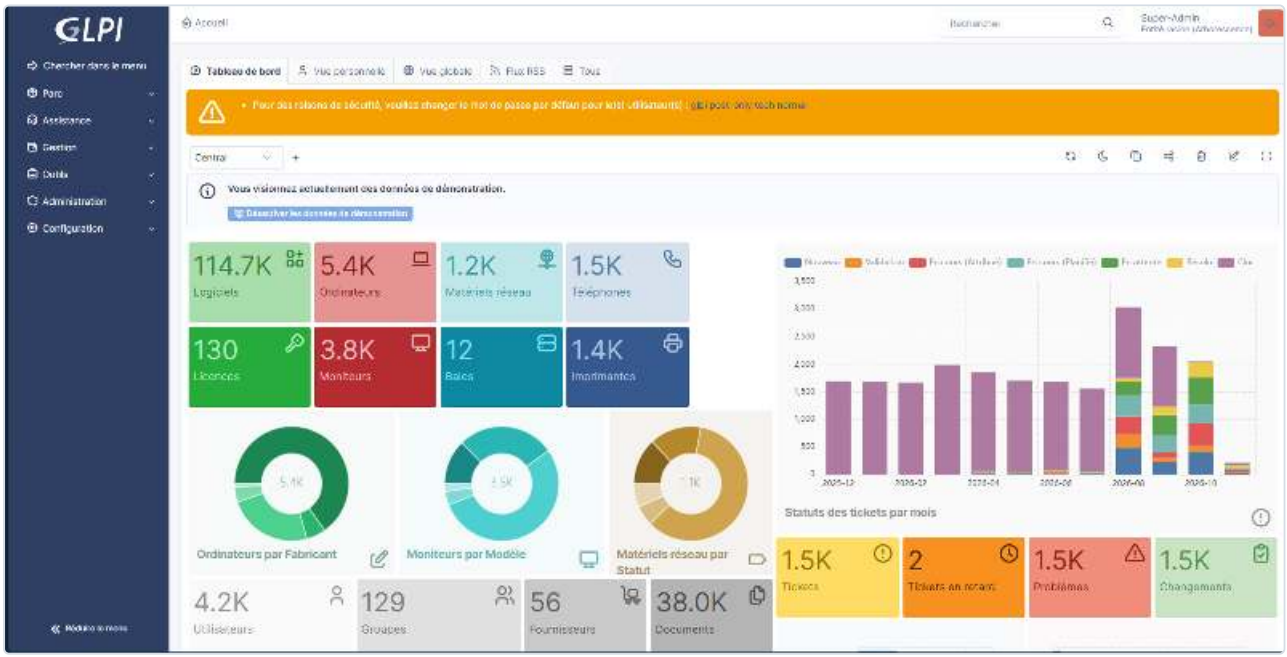

Figure 2 – Tableau de bord GLPI 11 après installation ; le bandeau orange rappelle les mots de passe par défaut à changer (glpi, post-only, tech, normal)

11 Sécuriser immédiatement après la première connexion

Les comptes par défaut sont publiquement documentés : les neutraliser avant tout usage réel.

- Changer le mot de passe du compte `glpi` (super-administrateur) par un secret unique.
- Changer le mot de passe, désactiver ou supprimer selon le besoin les comptes `tech`, `normal` et `post-only`.
- Créer un compte administrateur nominatif et limiter l'usage du compte intégré.

Supprimer le programme d'installation, qui n'a plus d'utilité et pourrait être détourné :

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

Attention

Le VirtualHost en port 80 convient au laboratoire. Ne jamais exposer cette configuration sur Internet : HTTPS, sauvegardes et mises à jour régulières sont indispensables en production.

4. Vérification

```
sudo -u www-data php /var/www/glpi/bin/console glpi:system:check_requirements
sudo apache2ctl configtest
sudo systemctl --no-pager status apache2 mariadb
curl -sI http://127.0.0.1/ | head -n 1
sudo mariadb -e "USE glpi; SHOW TABLES;" | wc -l
test ! -e /var/www/glpi/install/install.php && echo "install.php absent"
sudo tail -n 20 /var/log/apache2/glpi_error.log
```

Contrôle	Résultat attendu
Page d'accueil	GLPI répond via `SRV-GLPI.ngrange.local` ou `192.168.21.3`
Base	La base `glpi` contient les tables ; le compte SQL reste limité à `glpi.*`
Prérequis	Aucun prérequis obligatoire manquant
Apache	« Syntax OK » ; aucun dossier sensible n'est accessible par l'URL (par exemple `/config` ou `/files` retournent une erreur 404)

Résultat attendu

Le tableau de bord GLPI s'affiche : l'application, PHP, Apache et MariaDB communiquent correctement.

5. Dépannage

Symptôme	Cause probable	Solution
Page par défaut d'Apache au lieu de GLPI	Site `000-default` toujours actif ou `glpi.conf` non activé	`a2dissite 000-default.conf`, `a2ensite glpi.conf`, `systemctl restart apache2`
Erreur 404 sur toutes les pages sauf l'accueil	`mod_rewrite` non activé	`sudo a2enmod rewrite` puis redémarrer Apache
Extension manquante dans l'assistant	Paquet PHP non installé ou Apache non redémarré	Installer le paquet `php-…`, `systemctl restart apache2`, relancer `check_requirements`
Connexion SQL refusée	Mauvais mot de passe, utilisateur ou hôte	Tester `mariadb -u glpi -p -h localhost glpi` ; recréer l'utilisateur si besoin (ne pas utiliser root)
Écriture impossible dans config/ ou files/	Propriétaire incorrect	Réappliquer les `chown` de l'étape 6

6. Sécurité et bonnes pratiques

- **HTTPS** : certificat reconnu (Let's Encrypt ou PKI interne), redirection HTTP vers HTTPS, puis `session.cookie_secure = On` dans `/etc/php/8.5/apache2/php.ini`.
- **Cookies de session** : conserver `session.cookie_httponly = On` et `session.cookie_samesite = Lax` (ou plus strict).
- **Moindre privilège** : compte SQL limité à `glpi.*`, code en lecture seule pour Apache, pare-feu (par exemple `ufw`) limitant l'accès.
- **Tâches automatiques** : planifier le cron GLPI (`front/cron.php`, utilisateur `www-data`, chaque minute). Ajout au TP initial.
- **Sauvegarde** : exporter base et fichiers critiques, protéger les sauvegardes, tester la restauration.

```
# Adapter le dossier de sauvegarde et en protéger l'accès
sudo mariadb-dump --single-transaction glpi > /chemin-protege/glpi.sql
sudo tar -czf /chemin-protege/glpi-fichiers.tgz /var/www/glpi/config /var/www/glpi/files
```

- **Secrets** : jamais de mot de passe SQL en clair dans un script ou l'historique du shell.
- **Mises à jour** : suivre les publications GLPI, Ubuntu et plugins ; valider d'abord en test (ne pas rester sur un correctif antérieur à 11.0.12).

7. Aide-mémoire : utiliser GLPI au quotidien

Chemins de menu de l'interface française de GLPI 11. Les intitulés peuvent varier légèrement selon la version et la langue : en cas de doute, utiliser la zone « Chercher dans le menu » en haut à gauche.

7.1 Créer un utilisateur local

1 **Administration › Utilisateurs › « + Ajouter »**

- Champs : **Identifiant** (login), **Nom**, **Prénom**, **Mot de passe** (et confirmation), **Adresses de messagerie** (nécessaire aux notifications), **Profil** et **Entité** par défaut.
- Valider avec « Ajouter ». Les droits se précisent ensuite dans l'onglet **Habilitations** (voir 7.3).

7.2 Importer des utilisateurs depuis un annuaire LDAP / Active Directory

1 **Configuration › Authentification › Annuaires LDAP › « + Ajouter »**

- Champs importants : **Serveur** (contrôleur de domaine `SRV-AD1EX.ngrange.local` / `192.168.21.1`, réplica `SRV-AD2.ngrange.local` / `192.168.21.2`), **Port** (389, ou 636 en LDAPS), **BaseDN** (par exemple `DC=ngrange,DC=local` ou une OU, par exemple `OU=<NOM_OU>,DC=ngrange,DC=local`), **Filtre de connexion**, **RootDN** (compte de lecture dédié) et son mot de passe `<MOT_DE_PASSE>`, **Champ de l'identifiant** (`samaccountname` pour AD).
- Utiliser le bouton **Tester** de la fiche : la connexion doit être validée. L'extension `php-ldap` (installée à l'étape 2) est indispensable.
- Importer : Administration › Utilisateurs › « Lier un annuaire LDAP » › « Importation de nouveaux utilisateurs », rechercher puis cocher les comptes et importer.
- Pour attribuer automatiquement des profils/entités à l'import : Administration › Règles › « Règles d'affectation de droits à l'utilisateur ».

7.3 Profils, habilitations, entités et récursivité

1 **Administration › Entités › « + Ajouter » (créer une entité)**

- Champs : **Nom**, **Sous-entité de** (entité parente ; l'entité racine existe par défaut), coordonnées, adresse de messagerie de l'entité.
- Une entité isole un périmètre (site, client, service) : inventaire, tickets et utilisateurs y sont rattachés.

2 **Administration › Utilisateurs › [utilisateur] › onglet Habilitations › Ajouter une habilitation**

- Choisir le **Profil** (par exemple Super-Admin, Admin, Technician, Self-Service, Post-only, Observer) et l'**Entité**.
- **Récursif : Oui** donne les mêmes droits dans toutes les sous-entités ; **Non** limite à l'entité choisie.
- Un utilisateur peut cumuler plusieurs couples profil/entité ; l'un d'eux est le profil par défaut.
- Pour créer ou ajuster un profil : Administration › Profils (cocher les droits par module : lecture, création, modification, suppression).

7.4 Créer un ticket et gérer les catégories ITIL

1 **Assistance › Tickets › « + Ajouter »**

- Champs : **Type** (Incident ou Demande), **Catégorie**, **Urgence**, **Impact** (la **Priorité** est calculée à partir des deux), **Demandeur**, **Titre**, **Description**, éléments associés (ordinateur concerné), pièces jointes.
- Statuts usuels : Nouveau, En cours (attribué ou planifié), En attente, Résolu, Clos. Le suivi (réponses, solution, tâches) se saisit dans la fiche du ticket.

2 **Configuration › Intitulés › Assistance › Catégories ITIL (chemin à confirmer selon version)**

- Créer des catégories hiérarchiques (par exemple Matériel › Poste de travail) ; cocher les cases « Visible pour les incidents/demandes ». Elles servent aux statistiques et aux règles métier.

7.5 Assigner un technicien et créer des groupes

1 **Administration › Groupes › « + Ajouter », puis onglet Utilisateurs**

- Créer par exemple un groupe « Support N1 » et y ajouter les techniciens.
- Dans un ticket : section **Acteurs** › **Attribué à** › choisir un utilisateur et/ou un groupe, puis « Enregistrer ». Le statut passe à « En cours (attribué) ».

7.6 Inventaire automatique avec GLPI Agent

1 **Activer l'inventaire natif côté serveur**

- Administration › Inventaire › cocher **Activer l'inventaire** et enregistrer (chemin à confirmer selon version : Configuration › Générale › onglet Inventaire dans certaines versions).

2 **Installer l'agent sur un poste Linux**

Télécharger l'installateur Perl depuis la page Releases du dépôt GitHub `glpi-project/glpi-agent` (le nom contient la version) :

```
sudo perl glpi-agent-<VERSION>-linux-installer.pl --install --
server=http://192.168.21.3/ --runnow
sudo systemctl status glpi-agent
sudo glpi-agent --force      # forcer un inventaire immédiat
```

3 Installer l'agent sur un poste Windows

Installer le fichier MSI de la même page de téléchargement ; l'adresse du serveur peut aussi être saisie dans l'assistant graphique :

```
msiexec /i GLPI-Agent-<VERSION>-x64.msi /quiet SERVER="http://192.168.21.3/"
RUNNOW=1
```

- Utiliser le nom `SRV-GLPI.ngrange.local` à la place de l'IP seulement s'il est résolu par le DNS. Si l'inventaire n'arrive pas, tester l'URL `http://192.168.21.3/front/inventory.php` (point à valider selon version).
- Vérifier : page locale de l'agent `http://localhost:62354` ; dans GLPI : Parc › Ordinateurs, la machine apparaît après quelques minutes.

7.7 Ajouter un ordinateur manuellement

1 Parc › Ordinateurs › « + Ajouter »

- Champs : **Nom**, **Statut**, **Type**, **Fabricant**, **Modèle**, **Numéro de série**, **Lieu**, **Utilisateur** ou **Groupe**, **Entité**. Les composants (disques, mémoire) s'ajoutent dans les onglets de la fiche.

7.8 Règles métier

1 Administration › Règles › Règles métier pour les tickets › « + Ajouter »

- Définir le déclencheur (à l'ajout ou à la mise à jour), les **critères** (par exemple titre contient « imprimante ») et les **actions** (affecter la catégorie, le groupe technicien, la priorité).
- Autres familles de règles au même endroit : affectation d'un élément à une entité, affectation de droits aux utilisateurs importés. L'ordre des règles compte (glisser pour réordonner) ; tester sur des tickets fictifs.

7.9 Notifications par courriel (SMTP)

1 **Configuration › Notifications**

- Cocher **Activer les notifications par courriel** (et les suivis des tickets) puis ouvrir « Configuration des notifications par courriel ».
- Champs : **Courriel de l'administrateur**, **Courriel de l'expéditeur**, mode d'envoi **SMTP**, **Hôte**, **Port** (587 avec STARTTLS), **Sécurité**, **Identifiant SMTP** et mot de passe `<MOT_DE_PASSE>`.
- Cliquer sur **Envoyer un courriel de test**.
- Vérifier l'activation des modèles dans Configuration › Notifications › Notifications. Les courriels partent via une file d'attente traitée par l'action automatique `queuednotification` : le cron GLPI (section 6) doit donc tourner (Configuration › Actions automatiques).

7.10 Sauvegarder et restaurer

1 **Ligne de commande**

```
# Sauvegarde (voir section 6 pour les chemins)
sudo mariadb-dump --single-transaction glpi > /chemin-protege/glpi.sql
sudo tar -czf /chemin-protege/glpi-fichiers.tgz /var/www/glpi/config
/var/www/glpi/files

# Restauration sur une base glpi vide (après réinstallation de la pile)
sudo mariadb glpi < /chemin-protege/glpi.sql
sudo tar -xzf /chemin-protege/glpi-fichiers.tgz -C /
sudo chown -R www-data:www-data /var/www/glpi/config /var/www/glpi/files
```

- Conserver `config/glpicrypt.key` (dans `config`) : sans elle, les mots de passe chiffrés (SMTP, LDAP) ne sont plus lisibles.

7.11 Mettre à jour GLPI

1 Procédure de mise à jour d'une version corrective ou mineure

1. Sauvegarder base et fichiers (7.10), puis activer la maintenance si disponible : `sudo -u www-data php /var/www/glpi/bin/console glpi:maintenance:enable`.
2. Renommer l'ancien dossier : `sudo mv /var/www/glpi /var/www/glpi-old`, extraire la nouvelle archive dans `/var/www/`.
3. Recopier `config`, `files` (et `marketplace`) depuis l'ancienne installation, puis réappliquer les droits de l'étape 6.
4. Lancer la mise à jour de la base : ouvrir GLPI dans le navigateur (l'assistant de mise à jour s'affiche) ou en CLI `sudo -u www-data php /var/www/glpi/bin/console glpi:database:update` (nom de commande à confirmer avec `bin/console list`).
5. Tester, désactiver la maintenance (`glpi:maintenance:disable`), puis supprimer `glpi-old` après validation. Tester les plugins d'abord sur une copie.

7.12 Changer les mots de passe des comptes par défaut

1 Administration › Utilisateurs › [glpi, tech, normal, post-only]

- Ouvrir la fiche, saisir un nouveau **Mot de passe** et sa confirmation, puis « Sauvegarder ». Désactiver (case **Actif**) ou supprimer les comptes inutiles. Le bandeau orange du tableau de bord disparaît quand plus aucun mot de passe par défaut n'est utilisé.

7.13 Raccourcis et emplacements utiles

Élément	Valeur
URL	`http://192.168.21.3/` (ou `http://SRV-GLPI.ngrange.local/`)
Application	`/var/www/glpi` (public : `/var/www/glpi/public`)
Config GLPI (base, clé)	`/var/www/glpi/config/` (`config_db.php`, `glpicrypt.key`)
Données / documents	`/var/www/glpi/files/` (logs applicatifs dans `files/_log/` : `php-errors.log`, `sql-errors.log`)
VirtualHost	`/etc/apache2/sites-available/glpi.conf`
Logs Apache	`/var/log/apache2/glpi_error.log`, `glpi_access.log`
php.ini Apache	`/etc/php/8.5/apache2/php.ini`
Services	`sudo systemctl restart apache2 mariadb` ; `sudo apache2ctl configtest`
Console GLPI	`sudo -u www-data php /var/www/glpi/bin/console list`
Agent (poste)	Linux : `glpi-agent --force` ; page locale `http://localhost:62354`

Sources : documentation officielle GLPI (glpi-install.readthedocs.io) · dépôt glpi-project/glpi · notes de version Ubuntu 26.04 · TP réalisé en équipe.