

---

# GLPI Installation

GLPI Project, Teclib'

janv. 08, 2024



---

## Table des matières

---

<b>1</b>	<b>Pré-requis</b>	<b>3</b>
1.1	Serveur web . . . . .	3
1.1.1	Apache configuration . . . . .	3
1.1.2	Nginx configuration . . . . .	4
1.1.3	lighttpd configuration . . . . .	5
1.1.4	IIS configuration . . . . .	5
1.2	PHP . . . . .	6
1.2.1	Extensions obligatoires . . . . .	6
1.2.2	Extensions optionnelles . . . . .	6
1.3	Base de données . . . . .	7
<b>2</b>	<b>Installer GLPI</b>	<b>9</b>
2.1	Choisir une version . . . . .	9
2.2	Télécharger . . . . .	10
2.3	Installation . . . . .	10
2.4	Emplacements des dossiers et fichiers . . . . .	10
2.5	Après installation . . . . .	12
<b>3</b>	<b>Assistant d'installation</b>	<b>13</b>
3.1	Choix de la langue (Select your language) . . . . .	13
3.2	Licence . . . . .	14
3.3	Installation / mise à jour . . . . .	14
3.3.1	Vérification de l'environnement . . . . .	15
3.3.2	Connexion à la base de données . . . . .	16
3.3.3	Choix de la base de données . . . . .	16
3.3.4	Initialisation de la base de données . . . . .	17
3.3.5	Informations de télémétrie . . . . .	17
3.3.6	Installation terminée . . . . .	18
<b>4</b>	<b>Fuseaux horaires</b>	<b>19</b>
4.1	Utilisateurs non windows . . . . .	19
4.2	Utilisateurs windows . . . . .	20
4.3	Autoriser l'accès . . . . .	20
<b>5</b>	<b>Mise à jour</b>	<b>21</b>
<b>6</b>	<b>Outils en ligne de commande</b>	<b>23</b>

6.1	Options de la console . . . . .	23
6.2	Outils supplémentaires d'installation et mise à jour . . . . .	24
6.2.1	Vérification des prérequis . . . . .	24
6.2.2	Activer/Désactiver la maintenance . . . . .	24
6.3	Installation . . . . .	24
6.4	Configuration de la connexion à la base de données . . . . .	24
6.5	Mise à jour . . . . .	25
6.6	Clé de sécurité . . . . .	25
6.7	Autres outils . . . . .	26
6.7.1	Vérification du schéma de base de données . . . . .	26
6.7.2	Synchronisation LDAP . . . . .	26
6.7.3	Déverrouillage des tâches . . . . .	27
6.8	Outils pour les plugins . . . . .	27
6.9	Outils de migration . . . . .	28
6.9.1	De MyISAM à InnoDB . . . . .	28
6.9.2	Récupération des horodatages manquants . . . . .	28
6.9.3	Utilisation du type de données timestamp . . . . .	28
6.9.4	Migration du plugin Domaines . . . . .	28
6.9.5	Migration du plugin Racks . . . . .	29
<b>7</b>	<b>Configuration avancée</b> . . . . .	<b>31</b>
7.1	Connexion SSL à la base de données . . . . .	31

Cette documentation présente les instructions d'installation de **GLPI**.

GLPI (Gestion Libre de Parc Informatique) est un système libre (open source) de gestion de parc informatique et de helpdesk accessible via un navigateur web conçue pour gérer l'ensemble de vos problématiques de gestion de parc informatique, de la gestion de l'inventaire des composantes matérielles et logicielles d'un parc informatique à la gestion de l'assistance aux utilisateurs.



GLPI est une application web qui nécessite :

- un serveur web;
- PHP;
- une base de données.

## 1.1 Serveur web

GLPI requiert un serveur web qui supporte PHP, tel que :

- Apache 2 (ou plus récent);
- Nginx;
- lighttpd;
- Microsoft IIS.

### 1.1.1 Apache configuration

Here is a virtual host configuration example for Apache 2 web server.

**Avertissement :** The following configuration is only suitable for GLPI version 10.0.7 or later.

```
<VirtualHost *:80>
    ServerName glpi.localhost

    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
```

(suite sur la page suivante)

```

↳ NOT target the GLPI directory itself.
# Alias "/glpi" "/var/www/glpi/public"

<Directory /var/www/glpi/public>
    Require all granted

    RewriteEngine On

    # Ensure authorization headers are passed to PHP.
    # Some Apache configurations may filter them and break usage of API, CalDAV, ...
    RewriteCond %{HTTP:Authorization} ^(.+)$
    RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]

    # Redirect all requests to GLPI router, unless file exists.
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteRule ^(.*)$ index.php [QSA,L]
</Directory>
</VirtualHost>

```

**Note :** If you cannot change the Apache configuration (e.g. you are using a shared hosting), you can use a `.htaccess` file.

```

# /var/www/glpi/.htaccess
RewriteBase /
RewriteEngine On
RewriteCond %{REQUEST_URI} !^/public
RewriteRule ^(.*)$ public/index.php [QSA,L]

```

## 1.1.2 Nginx configuration

Here is a configuration example for Nginx web server using `php-fpm`.

**Avertissement :** The following configuration is only suitable for GLPI version 10.0.7 or later.

```

server {
    listen 80;
    listen [::]:80;

    server_name glpi.localhost;

    root /var/www/glpi/public;

    location / {
        try_files $uri /index.php$is_args$args;
    }

    location ~ ^/index\.php$ {

```

(suite sur la page suivante)

(suite de la page précédente)

```

# the following line needs to be adapted, as it changes depending on OS
↳distributions and PHP versions
fastcgi_pass unix:/run/php/php-fpm.sock;

fastcgi_split_path_info ^(.+\.(php|\.*)$);
include fastcgi_params;

fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
}
}

```

### 1.1.3 lighttpd configuration

Here is a virtual host configuration example for lighttpd web server.

**Avertissement :** The following configuration is only suitable for GLPI version 10.0.7 or later.

```

$HTTP["host"] =~ "glpi.localhost" {
    server.document-root = "/var/www/glpi/public/"

    url.rewrite-if-not-file = ( "" => "/index.php${url.path}${qsa}" )
}

```

### 1.1.4 IIS configuration

Here is a web.config configuration file example for Microsoft IIS. The physical path of GLPI web site must point to the public directory of GLPI (e.g. D:\glpi\public), and the web.config file must be placed inside this directory.

```

<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <system.webServer>
    <rewrite>
      <rules>
        <rule name="Rewrite to GLPI" stopProcessing="true">
          <match url="^(.*)$" />
          <conditions>
            <add input="{REQUEST_FILENAME}" matchType="IsFile" ignoreCase=
↳"false" negate="true" />
          </conditions>
          <action type="Rewrite" url="index.php" appendQueryString="true" />
        </rule>
      </rules>
    </rewrite>
  </system.webServer>
</configuration>

```

**Avertissement :** The [URL Rewrite](#) module is required.

## 1.2 PHP

TABLEAU 1 – PHP Compatibility Matrix

GLPI Version	Minimum PHP	Maximum PHP
9.5.X	7.2	8.0
10.0.X	7.4	8.2

---

**Note :** We recommend to use the newest supported PHP release for better performance.

---

### 1.2.1 Extensions obligatoires

Les extensions PHP suivantes sont requises pour que l'application fonctionne correctement :

- `dom`, `fileinfo`, `filter`, `libxml`, `json`, `simplexml`, `xmlreader`, `xmlwriter` : these PHP extensions are enable by default and are used for various operations ;
- `curl` : used for remote access to resources (inventory agent requests, marketplace, RSS feeds, ...);
- `gd` : used for images handling ;
- `intl` : used for internationalization ;
- `mysqli` : used for database connection ;
- `session` : used for sessions support ;
- `zlib` : used for handling of compressed communication with inventory agents, installation of gzip packages from marketplace and PDF generation.

### 1.2.2 Extensions optionnelles

---

**Note :** Bien que ces extensions ne soient pas requises, nous conseillons de les installer tout de même.

---

Les extensions PHP suivantes sont requises pour des fonctionnalités annexes de GLPI :

- `bz2`, `Phar`, `zip` : enable support of most common packages formats in marketplace ;
- `exif` : enhance security on images validation ;
- `ldap` : enable usage of authentication through remote LDAP server ;
- `openssl` : enable email sending using SSL/TLS ;
- `Zend OPcache` : enhance PHP engine performances.

## 1.3 Base de données

**Avertissement :** Currently, only [MySQL](#) (5.7 minimum) and [MariaDB](#) (10.2 minimum) database servers are supported by GLPI.

Pour fonctionner, GLPI nécessite un serveur de base de données.





Procéder comme suit :

1. *Configure your webserver*,
2. Choisir une version,
3. Télécharger l'archive,
4. Installer :)

### 2.1 Choisir une version

---

**Note :** Il est fortement recommandé de choisir la dernière version stable pour une utilisation en production.

---

GLPI suit une règle de versionnage sémantique, sur 3 chiffres. Le premier correspond à la version majeure, le second à la version mineure, le troisième à la version corrective.

Les version majeures peuvent amener d'importantes incompatibilités ainsi que de nouvelles fonctionnalités ; les versions mineures quant à elles apportent également de nouvelles fonctionnalités mais restent parfaitement compatibles au sein d'une version majeure.

Les versions correctives, quand à elles, se contentent de corriger les problèmes reportés sans ajout de fonctionnalités.

## 2.2 Télécharger

**Avertissement :** Sur GitHub, il existe systématiquement deux archives nommées *Source code* ; ces versions ne sont pas destinées à être téléchargées par les utilisateurs.

Aller à la section *téléchargement* du site web GLPI (ou obtenir l'archive directement depuis les [releases Github](#)) et choisir l'archive `glpi-{version}.tgz`.

## 2.3 Installation

L'installation à proprement parler de GLPI est composée de trois étapes :

1. Décompresser l'archive dans l'arborescence de votre serveur web ;
2. Donner les droits en écriture à votre serveur web sur les dossiers `files` et `config` ;
3. *lancer l'assistant d'installation* (ou utiliser *le script d'installation en ligne de commande*).

Une fois ces étapes effectuées l'application est prête à être utilisée.

Si vous avez besoin de configuration avancée, comme la mise en place de paramètres de connexion SSL, référez-vous à *la configuration avancée*.

## 2.4 Emplacements des dossiers et fichiers

Comme beaucoup d'applications web, GLPI peut être installé en copiant simplement le contenu complet du répertoire sur un serveur web quelconque. Néanmoins, une telle pratique peut s'avérer relativement peu sécurisée.

**Avertissement :** Tout fichier accessible directement depuis un serveur web doit être considéré comme non sécurisé !

GLPI stocke un certain nombre de données dans son répertoire `files`, la configuration de votre base de données est stockée dans le dossier `config`, ... Bien que des dispositions soient prises pour éviter que des données sensibles puisse être rendues accessibles directement depuis un serveur web, la meilleure façon de les protéger reste de les stocker en dehors de la racine web. De cette manière, les fichiers sensibles ne seront pas accessibles directement depuis le serveur web.

Un certain nombre de variables de configuration qu'il est possible d'utiliser pour ce faire (utilisées d'ailleurs dans les paquets fournis par les distributions linux) :

- `GLPI_CONFIG_DIR` : définit le chemin d'accès du dossier de configuration ;
- `GLPI_VAR_DIR` : définit le chemin d'accès du répertoire "fichiers" ;
- `GLPI_LOG_DIR` : définit le chemin d'accès vers les fichiers de log.

**Note :** Il existe bien d'autres variables de configuration, celles évoquées ci-dessus sont les principales à prendre en considération pour une installation davantage sécurisée.

Le choix des différents dossiers est laissé à votre entière appréciation ; l'exemple qui suit se base quand à lui sur les recommandations *FHS* `<http://www.pathname.com/fhs/>`.

Notre instance de GLPI sera installée dans le dossier `/var/www/glpi`, un hôte virtuel dans la configuration du serveur web reflètera ce chemin.

La configuration de GLPI sera stockée dans `/etc/glpi/`, copier simplement le contenu du dossier `config` à cet endroit. GLPI requiert un accès en lecture sur ce dossier pour fonctionner, et un accès en écriture lors de l'installation.

Les données de GLPI seront stockées dans `/var/lib/glpi/`, copier le contenu du dossier `files` à cet endroit. GLPI requiert les droits en lecture et écriture sur ce dossier.

Les fichiers de log de GLPI seront stockés dans `/var/log/glpi/`, il n'y a rien à copier. GLPI requiert les droits en lecture et écriture sur ce dossier.

Suivant cette nomenclature, il convient de créer un fichier `inc/downstream.php` dans le dossier de GLPI avec le contenu suivant :

```
<?php
define('GLPI_CONFIG_DIR', '/etc/glpi/');

if (file_exists(GLPI_CONFIG_DIR . '/local_define.php')) {
    require_once GLPI_CONFIG_DIR . '/local_define.php';
}
```

**Avertissement :** Les paquets GLPI des différents dépôts fourniront très certainement un fichier `inc/downstream.php`. Ce dernier ne doit pas être modifié !

GLPI looks for a *local\_define.php* file in its own *config* directory. If you want to use one from new config directory, you have to load it.

Ensuite, créer un fichier `/etc/glpi/local_define.php` avec le contenu suivant :

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

**Note :** Nouveau dans la version 9.2.2.

Pour les versions de GLPI antérieures à 9.2.2, la constante `GLPI_VAR_DIR` n'existait pas. Il fallait donc redéfinir tous les chemins un par un :

```
<?php
define('GLPI_VAR_DIR', '/var/lib/glpi');
define('GLPI_DOC_DIR', GLPI_VAR_DIR);
define('GLPI_CRON_DIR', GLPI_VAR_DIR . '/_cron');
define('GLPI_DUMP_DIR', GLPI_VAR_DIR . '/_dumps');
define('GLPI_GRAPH_DIR', GLPI_VAR_DIR . '/_graphs');
define('GLPI_LOCK_DIR', GLPI_VAR_DIR . '/_lock');
define('GLPI_PICTURE_DIR', GLPI_VAR_DIR . '/_pictures');
define('GLPI_PLUGIN_DOC_DIR', GLPI_VAR_DIR . '/_plugins');
define('GLPI_RSS_DIR', GLPI_VAR_DIR . '/_rss');
define('GLPI_SESSION_DIR', GLPI_VAR_DIR . '/_sessions');
define('GLPI_TMP_DIR', GLPI_VAR_DIR . '/_tmp');
define('GLPI_UPLOAD_DIR', GLPI_VAR_DIR . '/_uploads');
define('GLPI_CACHE_DIR', GLPI_VAR_DIR . '/_cache');

define('GLPI_LOG_DIR', '/var/log/glpi');
```

Of course, it is always possible to redefine any of those paths **if** needed.

## 2.5 Après installation

Une fois GLPI installé, vous avez presque terminé.

An extra step would be to secure installation directory. As an example, you can consider adding the following to your Apache virtual host configuration (or in the `glpi/install/.htaccess` file) :

```
<IfModule mod_authz_core.c>
  Require local
</IfModule>
<IfModule !mod_authz_core.c>
  order deny, allow
  deny from all
  allow from 127.0.0.1
  allow from ::1
</IfModule>
ErrorDocument 403 "<p><b>Restricted area.</b><br />Only local access allowed.<br />Check_
↪ your configuration or contact your administrator.</p>"
```

Avec cet exemple, l'accès au dossier `install` serait limité au seul hôte local et afficherait un message sinon. Bien sûr, vous pouvez adapter cela à vos besoins, référez-vous à la documentation de votre serveur web.



Pour débiter l'installation, faire pointer le navigateur web sur le dossier racine de GLPI : [https://\{adresse\\_glpi\}/](https://\{adresse_glpi\}/)  
Lorsque GLPI n'est pas installé; un processus d'installation étape par étape commence.

### 3.1 Choix de la langue (Select your language)

La première étape permet de choisir la langue dans laquelle se déroulera l'installation. Sélectionner la langue souhaitée, puis valider.



### 3.2 Licence

L'utilisation de GLPI est soumise à l'acceptation de la licence d'utilisation GNU. Une fois la licence lue et acceptée, valider le formulaire.



Sans acceptation des termes de la licence, il est impossible d'accéder aux étapes suivantes.

### 3.3 Installation / mise à jour

Cet écran permet de choisir entre installer GLPI ou effectuer une mise à jour.



Cliquer sur Installation.

### 3.3.1 Vérification de l'environnement

Cette étape va vérifier que le système satisfait les pré-requis. Si ce n'est pas le cas, il est impossible de continuer et un message d'erreur explicite indique les actions à réaliser avant d'essayer à nouveau.

**GLPI SETUP**

Step 0

Checking of the compatibility of your environment with the execution of GLPI

Test done	Results
Testing PHP Parser	✓
Sessions test	✓
Test if Session_use_trans_sid is used	✓
mysqli extension test	✓
ctype extension test	✓
fileinfo extension test	✓
json extension test	✓
mbstring extension test	✓
zlib extension test	✓
curl extension test	✓
gd extension test	✓
simplexml extension test	✓
xml extension test	✓
imap extension test	✓
APCu extension test	✓
xmldrpc extension test	✓
ldap extension test	⚠ ldap extension is not present
Zend OPcache extension test	⚠ Zend OPcache extension is not present
Allocated memory test	✓
Checking write permissions for setting files	✓
Checking write permissions for document files	✓
Checking write permissions for dump files	✓
Checking write permissions for session files	✓
Checking write permissions for automatic actions files	✓
Checking write permissions for graphic files	✓
Checking write permissions for lock files	✓
Checking write permissions for plugins document files	✓
Checking write permissions for temporary files	✓
Checking write permissions for cache files	✓
Checking write permissions for rss files	✓
Checking write permissions for upload files	✓
Checking write permissions for pictures files	✓
Checking write permissions for log files	✓
SELinux mode is Enforcing	✓
SELinux boolean configuration for httpd_can_network_connect --> on	✓
SELinux boolean configuration for httpd_can_network_connect_db --> on	✓
SELinux boolean configuration for httpd_can_sendmail --> on	✓

Do you want to continue?

[Continue](#) [Try again](#)

Certains pré-requis sont optionnels, il sera alors possible de valider le formulaire même s'ils ne sont pas respectés.

### 3.3.2 Connexion à la base de données

Les paramètres de connexion à la base de données sont demandés.



- *Serveur MySQL* : saisir le chemin de MySQL, par exemple *localhost*, ou *mysql.domaine.tld* ;
- *Utilisateur MySQL* : saisir le nom d'utilisateur ayant le droit de se connecter au serveur MySQL ;
- *Mot de passe MySQL* : saisir le mot de passe associé à l'utilisateur.

Une fois que ces champs sont correctement remplis, valider le formulaire.

Une première connexion à la base est alors établie. Si les paramètres sont invalides, un message d'erreur est affiché, il faut modifier les paramètres de connexion et essayer à nouveau.

### 3.3.3 Choix de la base de données

Une fois la connexion au serveur de base de données établie, il faut créer ou choisir la base de données souhaitée pour GLPI et l'initialiser.



Deux possibilités sont offertes :

- Utiliser une base existante

Sélectionner cette base de données dans la liste des bases affichées. Valider pour initialiser la base de données.

**Avertissement :** Le contenu de la base de données sélectionnée sera détruit lors de l'initialisation de celle-ci.

— Créer une nouvelle base de données

Sélectionner *Créer une nouvelle base de données*, saisir le nom de la nouvelle base dans le champ prévu à cet effet puis valider pour créer la base de données.

**Avertissement :** L'utilisateur SQL doit être en mesure de créer de nouvelles bases de données pour que cette option fonctionne.

### 3.3.4 Initialisation de la base de données

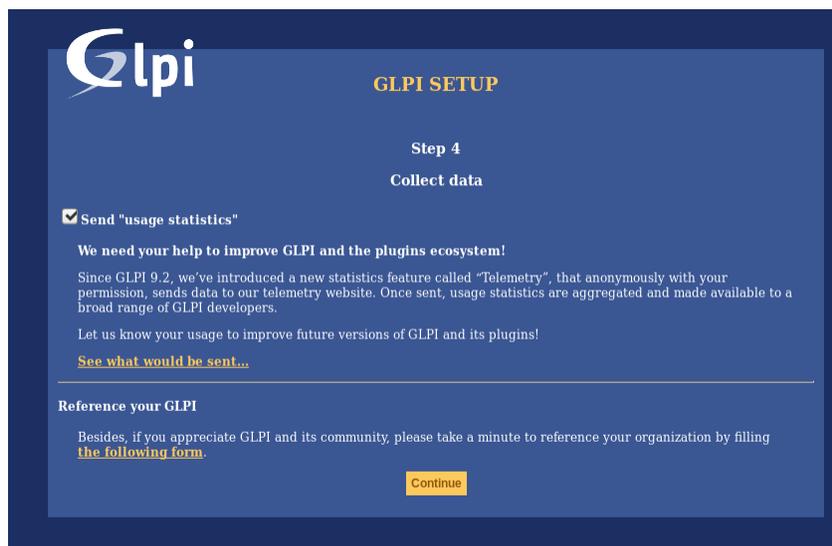
Cette étape initialise la base de données avec les valeurs par défaut.



En cas d'erreur, lire attentivement les informations affichées.

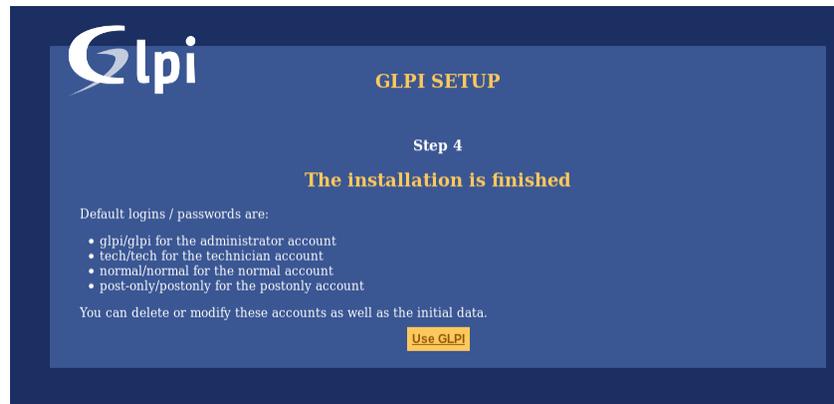
### 3.3.5 Informations de télémétrie

GLPI proposera le partage de quelques informations de télémétrie ainsi qu'un enregistrement. Ce n'est pas requis.



### 3.3.6 Installation terminée

Cette étape présente le récapitulatif de la procédure d'installation et donne la liste des comptes utilisateurs créés. Lire attentivement ces informations et valider pour effectuer la première connexion à l'application.



---

**Note :** Les comptes utilisateurs par défaut sont les suivants :

- *glpi/glpi* pour le compte administrateur,
  - *tech/tech* pour le compte technicien,
  - *normal/normal* pour le compte « normal »,
  - *post-only/postonly* pour le compte post-only.
- 

**Avertissement :** Pour des raisons de sécurité évidentes, supprimer ou modifier ces comptes.

Avant de supprimer le compte *glpi*, assurez-vous d'avoir créé un autre utilisateur avec le profil *super-admin*.



# CHAPITRE 4

---

## Fuseaux horaires

---

Afin que les fuseaux horaires puissent fonctionner sur une instance MariaDB/MySQL, vous devez initialiser les données des fuseaux horaires, et autoriser donner le droit READ sur leur table à l'utilisateur de la base de données GLPI.

**Avertissement :** Activer le support des fuseaux horaires sur votre instance MySQL peut affecter d'autres bases dans la même instance ; faites attention !

**Avertissement :** Actuellement, MySQL et MariaDB ont une date maximum limitée à 2038-01-19 pour les champs utilisant le type `timestamp` !

### 4.1 Utilisateurs non windows

Sur la plupart des systèmes, vous devrez initialiser les données des fuseaux horaires depuis ceux du système :

```
mysql_tzinfo_to_sql /usr/share/zoneinfo | mysql -p -u root mysql
```

Vous pourrez vérifier voir la [documentation de MariaDB à propos de mysql\\_tzinfo\\_to\\_sql](#) ainsi que la documentation de votre système pour savoir où sont stockées les données (si elles ne sont pas dans `/usr/share/zoneinfo`).

N'oubliez pas de redémarrer le serveur de base de données une fois que la commande a été exécutée avec succès.

### 4.2 Utilisateurs windows

Windows ne fournit pas d'informations sur les fuseaux horaires, vous devrez les télécharger et les initialiser vous-même. Consultez la [documentation MariaDB](#) à propos des fuseaux horaires.

### 4.3 Autoriser l'accès

**Avertissement :** Prêtez attention à ne pas donner d'autorisations trop larges à l'utilisateur de base de données GLPI, les tables système de devraient **jamais** être rendues accessibles aux utilisateurs applicatifs.

Afin de pouvoir lister les fuseaux horaires disponibles, votre utilisateur base de données GLPI doit avoir un accès en lecture sur la table `mysql.time_zone_name`. Partant du principe que votre utilisateur est `glpi@localhost`, vous devriez lancer quelque chose comme :

```
GRANT SELECT ON `mysql`.`time_zone_name` TO 'glpi'@'localhost';  
FLUSH PRIVILEGES;
```



---

## Mise à jour

---

- Note :** Comme pour tout processus de mise à jour, une sauvegarde des données doit être effectuée avant toute tentative :
- **sauvegarde de votre base de données** ;
  - backup your *config* directory, especially for your GLPI key file (*config/glpi.key* or *config/glpicrypt.key*) which is randomly generated ;
  - backup your *files* directory, it contains users and plugins generated files, like uploaded documents ;
  - backup your *marketplace* and *plugins* directory.
- 

Here are the steps to update GLPI :

- Download latest GLPI version.
- Ensure the target directory is empty and extract files there.
- Restore the previously backed up *config*, *files*, *marketplace* and *plugins* directory.
- Then open the GLPI instance URI in your browser, or (recommended) use the *php bin/console db :update command line tool*.

**Avertissement :** As soon as a new version of GLPI files is detected, you will not be able to use the application until the update process has been done.

**Avertissement :** Ne pas essayer de restaurer une sauvegarde sur une base de données non vide (par exemple, une base partiellement migrée pour quelque raison que ce soit).

S'assurer que la base est vide avant de restaurer la sauvegarde et tenter de mettre à jour, refaire en cas d'échec.

---

**Note :** Update process will automatically disable your plugins.

---

---

**Note :** Since GLPI 10.0.1, you can use the *php bin/console db :check command line tool* before executing the update command. This will allow you to check the integrity of your database, and to identify changes to your database that

---

could compromise the update.

---



---

## Outils en ligne de commande

---

Depuis GLPI 9.2.2, des outils en ligne de commande sont fournis et disponibles dans le dossier `script` de l'archive. Sur les versions précédentes, ces outils étaient présents dans le dossier `tools` qui n'est pas officiel et donc non fourni dans l'archive de release.

Depuis GLPI 9.4.0, les outils en ligne de commande sont centralisés dans une application console (`bin/console`). L'appel à `php bin/console` depuis le dossier de GLPI affiche la liste des commandes disponibles.

---

**Note :** Si APCu est installé sur votre système, il se peut qu'il provoque des échecs en ligne de commande puisque sa configuration par défaut le désactive pour la ligne de commande. Pour changer cela, définissez `apc.enable_cli` à `on` dans votre fichier de configuration APCu ;

---

**Avvertissement :** Lors de l'utilisation des outils en ligne de commande, pensez à vérifier l'utilisateur avec lequel vous êtes actuellement connecté, ainsi que les permissions sur les fichiers et dossiers. Avec un mauvais utilisateur, les fichiers de journalisation, de cache, ou d'autres fichiers, pourraient être créés avec des droits qui ne permettraient pas à votre serveur web d'y accéder en lecture ou écriture.

### 6.1 Options de la console

Pour l'ensemble des commandes console, les options suivantes sont disponibles :

- `--config-dir=CONFIG-DIR` chemin vers le dossier de configuration à utiliser, relatif au chemin courant (requis seulement si un chemin personnalisé est utilisé)
- `-h`, `--help` affiche la commande d'aide
- `--lang=LANG` langue utilisée pour la sortie (la valeur par défaut est celle configurée dans GLPI, ou « `en_GB` »)
- `-n`, `--no-interaction` désactive les questions interactives
- `--no-plugins` désactive les plugins de GLPI pendant l'exécution de la commande
- `-q`, `--quiet` désactive la sortie de la commande
- `-v|vv|vvv`, `--verbose=VERBOSE` niveau de verbosité : 1 pour une sortie normale, 2 pour plus de verbosité, et 3 pour le débogage

## 6.2 Outils supplémentaires d'installation et mise à jour

### 6.2.1 Vérification des prérequis

Before installing or upgrading, requirements are automatically checked ; but you can run them separately and see state for all of them using the `php bin/console glpi:system:check_requirements` command.

### 6.2.2 Activer/Désactiver la maintenance

GLPI fourni un mode maintenance qui peut être activé avant une mise à jour et désactivé une fois celle-ci effectuée et son bon fonctionnement contrôlé.

Utilisez simplement les commandes `glpi:maintenance:enable` et `glpi:maintenance:disable`.

## 6.3 Installation

La commande `php bin/console db:install` permet l'installation de GLPI en ligne de commande.

Les options possibles pour cette commande sont :

- `-r, --reconfigure` pour autoriser l'écrasement d'une éventuelle configuration de base de données déjà existante
- `-f, --force` pour forcer l'exécution de l'installation même si la base de données n'est pas vide
- `-L, --default-language=DEFAULT_LANGUAGE` langue par défaut de GLPI (*en\_GB* par défaut)
- `-H, --db-host=DB_HOST` nom d'hôte ou adresse IP (*localhost* par défaut)
- `-P, --db-port=DB_PORT` port de la base de données (port par défaut de MySQL si l'option n'est pas définie)
- `-d, --db-name=DB_NAME` nom de la base de données
- `-u, --db-user=DB_USER` nom d'utilisateur de la base de données
- `-p, --db-password=DB_PASSWORD` database user's password (use it without value to be prompted for password)

Si les options obligatoires ne sont pas spécifiées lors de l'appel à la commande, la console vous invitera à les saisir.

Database connection parameters may be omitted if a configuration file already exists.

Voir aussi les *options de console*.

## 6.4 Configuration de la connexion à la base de données

Nouveau dans la version 9.5.0.

La commande `php bin/console db:configure` permet de définir les paramètres de connexion à la base de données en ligne de commande.

Les options possibles pour cette commande sont :

- `-r, --reconfigure` pour autoriser l'écrasement d'une éventuelle configuration de base de données déjà existante
- `-H, --db-host=DB_HOST` nom d'hôte ou adresse IP (*localhost* par défaut)
- `-P, --db-port=DB_PORT` port de la base de données (port par défaut de MySQL si l'option n'est pas définie)
- `-d, --db-name=DB_NAME` nom de la base de données
- `-u, --db-user=DB_USER` nom d'utilisateur de la base de données
- `-p, --db-password=DB_PASSWORD` database user's password (use it without value to be prompted for password)

Si les options obligatoires ne sont pas spécifiées lors de l'appel à la commande, la console vous invitera à les saisir.

Voir aussi les *options de console*.

## 6.5 Mise à jour

La commande `php bin/console db:update` permet de mettre à jour GLPI, depuis une précédente version, en ligne de commande.

Il n'y a pas d'arguments requis. Lancez simplement la commande pour procéder à la mise à jour de la base de données.

**Avertissement :** Ne pas oublier d'effectuer une sauvegarde de la base de données avant toute tentative de mise à jour !

**Avertissement :** Since GLPI 10.0.2, `db:check_schema_integrity` is executed before performing the update. If an error is detected, the command will ask you if you want to continue (unless `--no-interaction` is used). You can bypass this `db:check_schema_integrity` by using the option `-s, --skip-db-checks`.

Les options possibles pour cette commande sont :

- `-u, --allow-unstable` autorise la mise à jour vers une version instable (à utiliser avec précaution)
- `-f, --force` force l'exécution de la mise à jour depuis la version v-1, même si la base de données semble déjà être à jour
- `-s, --skip-db-checks` do not check database schema integrity before performing the update
- `--enable-telemetry` allow usage statistics sending to Telemetry service (<https://telemetry.glp-project.org>)
- `--no-telemetry` disallow usage statistics sending to Telemetry service (<https://telemetry.glp-project.org>)

Voir aussi les *options de console*.

## 6.6 Clé de sécurité

Nouveau dans la version 9.4.6.

---

**Note :** Le fichier de clé de GLPI est disponible pour GLPI  $\geq$  9.4.6 mais n'est pas obligatoire. À partir de GLPI 9.5, utilisé un fichier de clé sera obligatoire.

---

Dans le but de stocker certains contenus sensibles, GLPI se base sur un outil fait maison de chiffrement/déchiffrement, qui utilise une clé pour :

- chiffrer les données avant de les stocker en base de données,
- déchiffrer les données qui ont été récupérées en base de données.

La commande `php bin/console glpi:security:change_key` permet de changer la clé, par exemple si elle a été compromise. Par défaut, la commande va :

- générer une nouvelle clé et la stocker dans le fichier de clé,
- mettre à jour tous les champs configurés (pour le cœur et les plugins), afin d'utiliser la nouvelle clé,
- mettre à jour tous les éléments de configuration listés (pour le cœur et les plugins), afin d'utiliser la nouvelle clé,

## 6.7 Autres outils

### 6.7.1 Vérification du schéma de base de données

The `php bin/console db:check_schema_integrity` command can be used to check if your database schema differs from expected one.

Les options possibles pour cette commande sont :

- `--strict` : Strict comparison of definitions
- `--check-all-migrations` : Check tokens related to all databases migrations.
- `--check-innodb-migration` : Check tokens related to migration from « MyISAM » to « InnoDB ».
- `--check-timestamps-migration` : Check tokens related to migration from « datetime » to « timestamp ».
- `--check-utf8mb4-migration` : Check tokens related to migration from « utf8 » to « utf8mb4 ».
- `--check-dynamic-row-format-migration` : Check tokens related to « DYNAMIC » row format migration.
- `--check-unsigned-keys-migration` : Check tokens related to migration from signed to unsigned integers in primary/foreign keys.
- `-p, --plugin` : Plugin to check. If option is not used, checks will be done on GLPI core database tables.

Si vous avez une quelconque différence, la sortie ressemblera à ceci :

```
$ php bin/console glpi:database:check_schema_integrity
Table schema differs for table "glpi_rulecriterias".
--- Original
+++ New
@@ @@
 create table `glpi_rulecriterias` (
   `id` int(11) not null auto_increment
   `rules_id` int(11) not null default '0'
   `criteria` varchar(255) default null
   `condition` int(11) not null default '0'
- `pattern` text default null
+ `pattern` text
   primary key (`id`)
```

Compared to the GLPI installation file :

- a line that starts with `-` means that something is missing in your database
- a line that starts with `+` means that there is something extra in your database

You can also have a message like `Unknown table "glpi_tablename" has been found in database.`, this indicates that this table doesn't exist in the installation file of the current GLPI schema :

- either it's a table that you have voluntarily created for your needs, you can ignore this message
- either it's an old GLPI table which is no longer useful, you can delete it (taking care to make a backup before)

### 6.7.2 Synchronisation LDAP

La commande `bin/console glpi:ldap:synchronize_users` permet de synchroniser les utilisateurs de GLPI sur la base des informations d'un serveur LDAP.

Les options possibles pour cette commande sont :

- `-c, --only-create-new` créer seulement les nouveaux utilisateurs
- `-u, --only-update-existing` met à jour seulement les utilisateurs déjà existants
- `-s, --ldap-server-id[=LDAP-SERVER-ID]` ne synchronise que les informations associées au serveur LDAP désigné (plusieurs valeurs possibles)
- `-f, --ldap-filter[=LDAP-FILTER]` filtre à appliquer lors de la recherche LDAP
- `--begin-date[=BEGIN-DATE]` date de début à appliquer dans le filtre « modifyTimestamp »

- `--end-date[=END-DATE]` date de fin à appliquer dans le filtre « modifyTimestamp »
- `-d, --deleted-user-strategy[=DELETED-USER-STRATEGY]` force la stratégie utilisée pour les utilisateurs supprimés :
  - 0 : Préserver
  - 1 : Mettre dans la corbeille
  - 2 : Retirer des autorisations et des groupes dynamiques
  - 3 : Désactiver
  - 4 : Désactiver + Retirer des autorisations et des groupes dynamiques

Voir <https://www.php.net/manual/fr/datetime.formats.php> pour les formats de date supportés dans les options `--begin-date` et `--end-date`.

Voir aussi les *options de console*.

### 6.7.3 Déverrouillage des tâches

La commande `php bin/console task:unlock` peut être utilisée pour déverrouiller les tâche planifiées bloquées.

**Avvertissement :** Gardez à l'esprit qu'aucune tâche ne devrait rester bloquée, sauf de la cas d'un bug ou d'une défaillance système (une défaillance de base de données pendant l'exécution d'une tâche par exemple).

Les options possibles pour cette commande sont :

- `-a, --all` déverrouiller toutes les tâches
- `-c, --cycle[=CYCLE]` temps d'exécution (en cycles) à partir duquel une tâche est considérée comme bloquée (délai = fréquence d'exécution de la tâche \* cycle)
- `-d, --delay[=DELAY]` temps d'exécution (en secondes) à partir duquel une tâche est considérée comme bloquée (1800 secondes par défaut)
- `-t, --task[=TASK]` `itemtype::name` de la tâche à débloquent (par exemple : `MailCollector::mailgate`)

Voir aussi les *options de console*.

## 6.8 Outils pour les plugins

Nouveau dans la version 9.5.

Certaines commandes sont aussi disponibles pour gérer les plugins en ligne de commande :

- `glpi:plugin:install`
- `glpi:plugin:activate`
- `glpi:plugin:deactivate`

Pour installer le plugin `MyGreatPlugin`; vous devriez effectuer ceci :

```
$ ./bin/console glpi:plugin:install MyGreatPlugin
$ ./bin/console glpi:plugin:activate MyGreatPlugin
```

Chacune de ces commandes peut prendre comme argument le nom d'un plugin, ou l'option `--all` si vous souhaitez la lancer pour tous les plugins.

## 6.9 Outils de migration

### 6.9.1 De MyISAM à InnoDB

Nouveau dans la version 9.3.0.

Depuis la version 9.3.0, GLPI utilise le moteur InnoDB en remplacement du moteur MyISAM précédemment utilisé.

La commande `php bin/console glpi:migration:mysam_to_innodb` peut être utilisée pour migrer les tables existantes vers le moteur InnoDB.

### 6.9.2 Récupération des horodatages manquants

Nouveau dans la version 9.1.0.

Avant GLPI 9.1.0, les champs correspondant aux dates de création et de modification n'existaient pas.

La commande `php bin/console glpi:migration:build_missing_timestamps` peut être utilisée pour reconstruire les valeurs manquantes en utilisant les données de journalisation.

### 6.9.3 Utilisation du type de données timestamp

Nouveau dans la version 9.5.0.

De nombreux champs de date utilisaient le type de données DATETIME, mais celui-ci ne permet pas un usage des fuseaux horaires. Le support de ces fuseaux nécessite que tous les champs utilisent le type de données TIMESTAMP, mais la requête de mise à jour peut s'avérer très longue à exécuter et n'est donc pas incluse dans le processus de mise à jour standard.

Utiliser la commande `glpi:migration:timestamps` changera ces champs pour qu'ils utilisent le bon type de données, mais pensez à lire la [documentation relative aux fuseaux horaires](#) avant d'effectuer la migration.

<b>Avertissement :</b> Assurez vous de faire une sauvegarde de votre base données avant l'opération !
---

### 6.9.4 Migration du plugin Domaines

Nouveau dans la version 9.5.0.

Dans GLPI, les domaines ont évolué pour passer d'un simple intitulé à un objet plus complexe, incluant, entre autres, la gestion des enregistrements de noms. Ainsi, les fonctionnalités du plugin Domaines sont maintenant présentes dans le cœur de GLPI.

Pour migrer les données du plugin, utilisez la commande `glpi:migration:domains_plugin_to_core`. La présence du plugin est nécessaire pour permettre certaines vérifications, mais vous pouvez utiliser l'option `--without-plugin` pour désactiver celles-ci, même si nous vous le déconseillons. Si vous utilisiez précédemment une ancienne version du plugin, vous pouvez utiliser l'option `--update-plugin` pour que le plugin mette ses données à niveau.

Après l'opération, tous les types de domaines, les domaines, ainsi que les relations avec d'autres objets auront été migrés dans les tables du cœur de GLPI.

## 6.9.5 Migration du plugin Racks

Nouveau dans la version 9.5.0.

Depuis GLPI 9.3.0, la gestion d'infrastructure des centres de données est disponible au sein des fonctionnalités du cœur de GLPI. Un script de migration des données du plugin a été fourni dans le dossier `scripts`. Depuis GLPI 9.5.0, cette migration a été revue et a été déplacée au sein de la console de commandes.

Pour migrer les données du plugin, utilisez la commande `glpi:migration:racks_plugin_to_core`. La présence du plugin est nécessaire pour permettre certaines vérifications, mais vous pouvez utiliser l'option `--without-plugin` pour désactiver celles-ci, même si nous vous le déconseillons. Si vous utilisiez précédemment une ancienne version du plugin, vous pouvez utiliser l'option `--update-plugin` pour que le plugin mette ses données à niveau.





### 7.1 Connexion SSL à la base de données

Nouveau dans la version 9.5.0.

Une fois l'installation terminée, vous pouvez mettre à jour le fichier `config/config_db.php` pour définir les paramètres de connexion SSL. Les paramètres disponibles correspondent à `mysqli::ssl_set()` :

- `$dbssl` défini si la connexion doit utiliser SSL (*false* par défaut)
- `$dbsslkey` chemin vers le fichier contenant la clé (*null* par défaut)
- `$dbsslcert` chemin vers le fichier contenant le certificat (*null* par défaut)
- `$dbsslca` chemin vers le fichier contenant l'autorité du certificat (*null* par défaut)
- `$dbsslcapath` chemin vers le dossier contenant les certificats SSL CA au format PEM (*null* par défaut)
- `$dbsslcacipher` liste des chiffres autorisés à être utilisés pour le chiffage SSL (*null* par défaut)

**Avertissement :** Pour le moment, il n'est pas possible de définir les paramètres de connexion SSL avant ou pendant le processus d'installation. Ceci doit être fait une fois l'installation effectuée.

