
GLPI Installation

GLPI Project, Teclib'

20.03.2024

1	Technické náležitosti	3
1.1	Webový server	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	Interpret PHP	6
1.2.1	Nezbytná rozšíření	6
1.2.2	Volitelná rozšíření	6
1.2.3	Security configuration for sessions	7
1.3	Databáze	7
2	Instalace GLPI	9
2.1	Volba verze	9
2.2	Stažení	10
2.3	Instalace	10
2.4	Umístění souborů a složek	10
2.5	Po instalaci	12
3	Průvodce instalací	13
3.1	Zvolte jazyk (vyberte svůj jazyk)	13
3.2	Licence	13
3.3	Instalace/aktualizace	14
3.3.1	Kontroly prostředí	14
3.3.2	Připojení do databáze	15
3.3.3	Volba databáze	16
3.3.4	Inicializace databáze	17
3.3.5	Telemetrické údaje	17
3.3.6	Závěr instalace	18
4	Časové zóny	19
4.1	Non windows users	19
4.2	Windows users	20
4.3	Udělit přístup	20
5	Aktualizace	21

6	Nástroje pro příkazový řádek	23
6.1	Console options	23
6.2	Additional install and update tools	24
6.2.1	Kontrola požadavků	24
6.2.2	Zapnout/Vypnout režim údržby	24
6.3	Instalace	24
6.4	Nastavení připojení k databázi	24
6.5	Aktualizace	25
6.6	Bezpečnostní klíč	25
6.7	Various tools	26
6.7.1	Kontrola databázového schématu	26
6.7.2	LDAP synchronizace	27
6.7.3	Task unlock	27
6.8	Plugins tools	28
6.9	Migration tools	28
6.9.1	From MyISAM to InnoDB	28
6.9.2	Missing timestamps builder	28
6.9.3	Use timestamp data type	28
6.9.4	Migrate Domains plugin	29
6.9.5	Migrate Racks plugin	29
7	Pokročilé nastavení	31
7.1	SSL připojení k databázi	31

Tento dokument představuje pokyny pro instalaci [GLPI](#).

GLPI (Gestion Libre de Parc Informatique – svobodná správa počítačového vybavení) je open source řešení správy majetku a služby podpory (helpdesk). Ovládá se přes webový prohlížeč a je navrženo pro správu veškerých záležitostí ohledně počítačových zařízení – od hardware, přes soupis software až po správu služby podpory.

GLPI je webová aplikace, která bude potřebovat:

- webový server;
- interpret PHP;
- databázi.

1.1 Webový server

GLPI potřebuje webový server který spolupracuje s PHP, jako například:

- Apache 2 (nebo novější);
- Nginx;
- lighttpd;
- Microsoft IIS.

1.1.1 Apache configuration

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

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

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

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

(continues on next page)

(pokračujte na předchozí stránce)

```
# If you want to place GLPI in a subfolder of your site (e.g. your virtual host is
↳serving multiple applications),
# you can use an Alias directive. If you do this, the DocumentRoot directive MUST
↳NOT target the GLPI directory itself.
# Alias "/glpi" "/var/www/glpi/public"

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

    RewriteEngine On

    # 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>
```

Poznámka: 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.

Varování: 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;
```

(continues on next page)

(pokračujte na předchozí stránce)

```

}

location ~ ^/index\.php$ {
    # 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.

Varování: 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>

```

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

1.2 Interpret PHP

Tabulka 1: PHP Compatibility Matrix

GLPI Version	Minimum PHP	Maximum PHP
10.0.X	7.4	8.3

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

1.2.1 Nezbytná rozšíření

Aby aplikace fungovala správně, jsou potřebná následující rozšíření pro PHP:

- `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 Volitelná rozšíření

Poznámka: I když tato rozšíření nejsou povinná, je doporučeno je nainstalovat i tak.

Pro některé další funkce GLPI jsou potřebná následující rozšíření pro PHP:

- `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.

1.2.3 Security configuration for sessions

To enhance security, it is recommended to configure PHP sessions with the following settings:

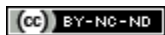
- `session.cookie_secure`: should be set to **on** when GLPI can be accessed on **only** HTTPS protocol;
- `session.cookie_httponly`: should be set to **on** to prevent client-side scripts from accessing cookie values;
- `session.cookie_samesite`: should be set, at least, to **Lax**, to prevent cookies from being sent cross-origin (across domains) POST requests.

Poznámka: Refer to [PHP documentation](#) for more information about session configuration.

1.3 Databáze

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

Aby mohlo fungovat, GLPI vyžaduje databázový server.



Postupujte následovně:

1. *Configure your webserver*,
2. Zvolte verzi,
3. Stáhněte si archiv,
4. Nainstalujte :)

2.1 Volba verze

Poznámka: Pro použití v produkčním prostředí je důrazně doporučeno používat nejnovější stabilní vydání.

GLPI používá významové schéma označování verzí, se třemi číslicemi. První je hlavní vydání, druhé dílčí a třetí opravné.

Hlavní vydání mohou přinášet nové funkce, ale z důvodu podstatných změn také vyhnutelné nekompatibility; to dílčí verze mohou přinášet spíše drobnější nové funkce, zato v rámci daného hlavního vydání zůstává kompatibilita zachována.

Opravná vydání řeší pouze nahlášené problémy a nepřidávají nic nového.

2.2 Stažení

Varování: Na portálu GitHub jsou vždy ještě dva další archivy nazvané *Source code* (zdrojové kódy) – ty nepoužívejte.

Jděte do sekce *download* (stahování) [webových stránek GLPI](#) (nebo získejte archiv přímo z [vydání na portálu Github](#)) a zvolte archiv `glpi-{verze}.tgz`.

2.3 Instalace

Instalace GLPI samotná se skládá ze tří kroků:

1. Rozbalení archivu do kořenové složky vašeho webu;
2. Udělení práv k zápisu do složek `files` a `config` vašemu webovému serveru;
3. Spuštění *průvodce instalací* (nebo použití *skriptu pro instalaci z příkazového řádku*).

Po dokončení těchto třech kroků je aplikace připravena k používání.

Pokud potřebujete nastavit rozšířenou konfiguraci, například parametry připojení SSL, přejděte na: doc: *pokročilá konfigurace* `<advanced-configuration>`.

2.4 Umístění souborů a složek

Jako mnoho ostatních webových aplikací, GLPI je možné nainstalovat pouhým zkopírováním celé složky na libovolný webový server. Nicméně to může být méně zabezpečené.

Varování: Každý soubor přístupný přímo z webového serveru je třeba považovat za nezabezpečený!

GLPI uchovává některá data ve složce `files`, přístupové údaje k databázi ve složce `config`, atd. I když GLPI poskytuje některé způsoby jak zabránit, aby k souborům mohl webový server přímo přistupovat, nejlepší je uchovávat je mimo kořenovou složku webu. Tím nemůže být k souborům s citlivým obsahem přistupováno přímo prostřednictvím webového serveru.

Je k dispozici několik direktiv nastavení, kterými je toho možné docílit (direktivy které jsou použity v balíčcích, poskytovaných linuxovými distribucemi):

- `GLPI_CONFIG_DIR`: nastavit popis umístění složky s nastaveními;
- `GLPI_VAR_DIR` : nastavit popis umístění složky `files`;
- `GLPI_LOG_DIR`: nastavit popis umístění souborů pro záznam událostí (log).

Poznámka: K dispozici je mnoho dalších direktiv nastavení, ty které zde byly zmíněny jsou hlavní, které mít na paměti pro lépe zabezpečenou instalaci.

Volba složek je zcela na vás. Následující příklad dodržuje doporučení standardu [FHS](#) .

Uváděná GLPI instance bude nainstalována do `/var/www/glpi` a pro toto umístění bude webovém serveru nastaven virtuální hostitel.

Konfigurace GLPI bude uložena v `/etc/glpi`, zkopírujte obsah adresáře `config` na toto místo. GLPI vyžaduje práva čtení v tomto adresáři, aby fungoval; a práva na zápis během procesu instalace.

Data GLPI budou uložena v `/var/lib/glpi`, stačí zkopírovat obsah adresáře `files` na toto místo. GLPI vyžaduje práva pro čtení a zápis v tomto adresáři.

Záznamy události (log) v GLPI budou ukládány do `/var/log/glpi`. Není co kopírovat – stačí jen tuto složku vytvořit. GLPI potřebuje práva na čtení a zápis do této složky.

Dle těchto pokynů, vytvoříme ve složce GLPI soubor `inc/downstream.php` s následujícím obsahem:

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

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

Varování: Balíčky s GLPI v rámci linuxové distribuce určitě poskytnou soubor `inc/downstream.php` – neupravujte ho!

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.

Pak vytvoříme soubor `/etc/glpi/local_define.php` s následujícím obsahem:

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

Poznámka: Nové ve verzi 9.2.2.

V GLPI před verzí 9.2.2 neexistovala proměnná `GLPI_VAR_DIR` a bylo třeba nastavovat všechny popisy umístění zvlášť:

```
<?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.

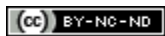
2.5 Po instalaci

Poté, co je GLPI nainstalované, je téměř hotovo.

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>"
```

V této ukázce bude složka *install* přístupná pouze z počítače samotného a jinak zobrazí chybu. Samozřejmě si toto můžete přizpůsobit svým potřebám. Nahlédněte do dokumentace k webovému serveru, který používáte.

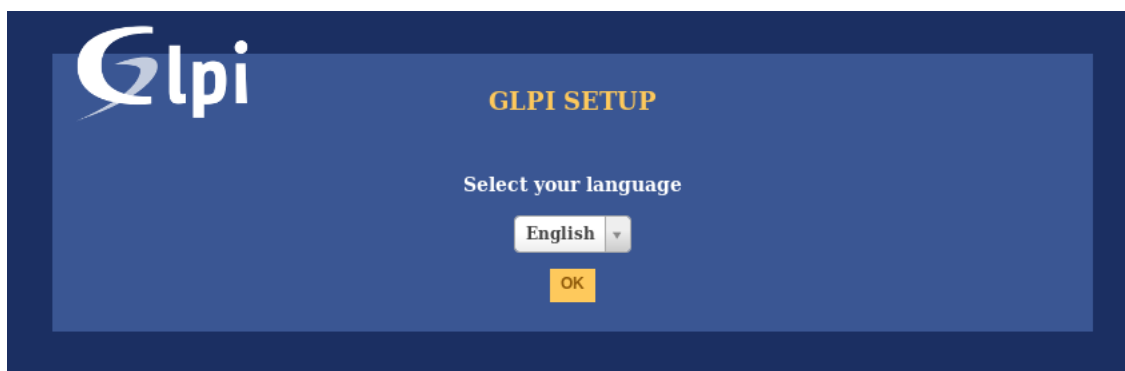


Pro zahájení instalace nasměrujte svůj webový prohlížeč na hlavní adresu GLPI: https://\{adresa_glpi\}/

Pokud GLPI ještě není nainstalováno, jsou zahájeny jednotlivé kroky instalačního procesu.

3.1 Zvolte jazyk (vyberte svůj jazyk)

Prvním krokem je volba jazyka instalace. Vyberte svůj jazyk a potvrďte.



3.2 Licence

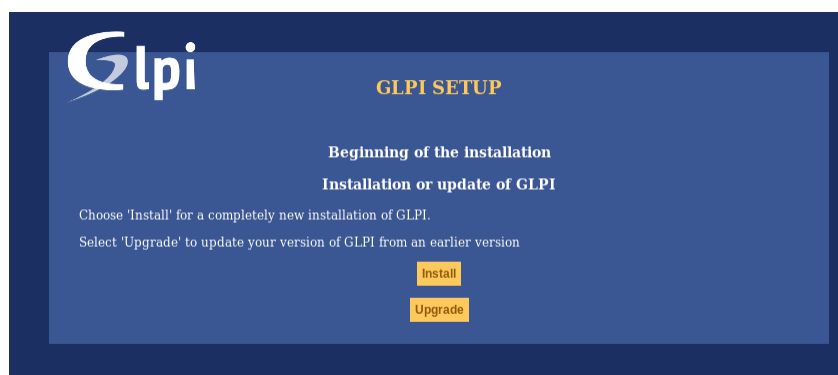
Používání GLPI vyžaduje souhlas s licencí GNU GPL (zaručuje svobody). Po přečtení a souhlasu s licenčními podmínkami potvrďte.



Pokud nesouhlasíte s podmínkami licence, není možné pokračovat v instalaci.

3.3 Instalace/aktualizace

Na této obrazovce je možné zvolit mezi novou instalací GLPI a aktualizací.



Klikněte na Nainstalovat.

3.3.1 Kontroly prostředí

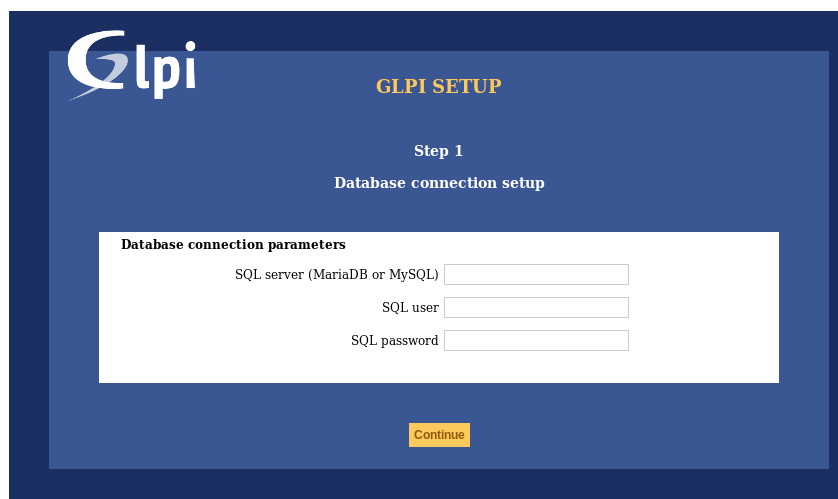
Tento krok zkontroluje, zda jsou splněny všechny předpoklady. Pokud tomu tak není, není možné pokračovat a explicitní chybová zpráva vám řekne, co je špatného a co dělat, než to zkusíte znovu.



Některé náležitosti jsou volitelné, v instalaci bude možné pokračovat i když nejsou splněné.

3.3.2 Připojení do databáze

Jsou vyžádány parametry pro připojení do databáze.



GLPI

GLPI SETUP

Step 1

Database connection setup

Database connection parameters

SQL server (MariaDB or MySQL)

SQL user

SQL password

Continue

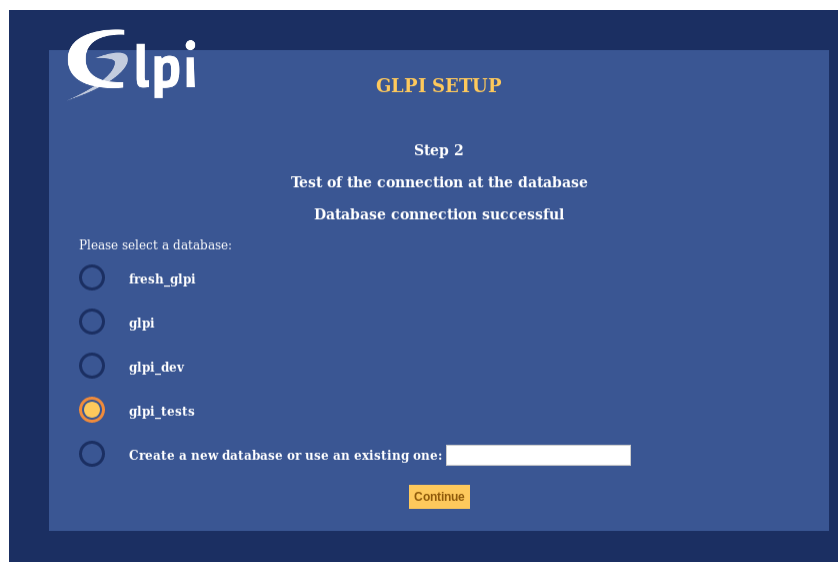
- *SQL server*: zadejte popis umístění svého SQL serveru, například *localhost* nebo *mysql.domenovynazev.tld*;
- *SQL uživatel*: zadejte uživatelské jméno účtu, který je oprávněn se připojovat do databáze;
- *SQL heslo*: zadejte heslo uživatele.

Po řádném vyplnění všech kolonek, klikněte na Pokračovat.

Poté je navázáno první spojení do databáze. Pokud parametry nejsou platné, zobrazí se chybové hlášení a bude třeba parametry opravit – a pak vyzkoušet znovu.

3.3.3 Volba databáze

Po úspěšném navázání spojení s databázovým serverem je třeba zadat vytvoření nebo zvolit existující databázi, kterou chcete použít pro GLPI a inicializovat ji.



GLPI

GLPI SETUP

Step 2

Test of the connection at the database

Database connection successful

Please select a database:

☐ fresh_glpi

☐ glpi

☐ glpi_dev

☒ glpi_tests

☐ Create a new database or use an existing one:

Continue

Pokračovat je tedy možné dvěma způsoby:

- Použít existující databázi

Vyberte tuto databázi ze zobrazeného seznamu. Klikněte na Pokračovat a potvrďte.

Varování: Při instalaci bude jakýkoli případný obsah vybrané databáze zlikvidován.

- Vytvořit novou databázi

Zvolte *Vytvořit novou databázi*, do odpovídající kolonky pro ní zadejte název a klikněte na Pokračovat.

Varování: Aby tato volba fungovala je třeba, aby daný SQL uživatel měl práva na vytváření nových databází.

3.3.4 Inicializace databáze

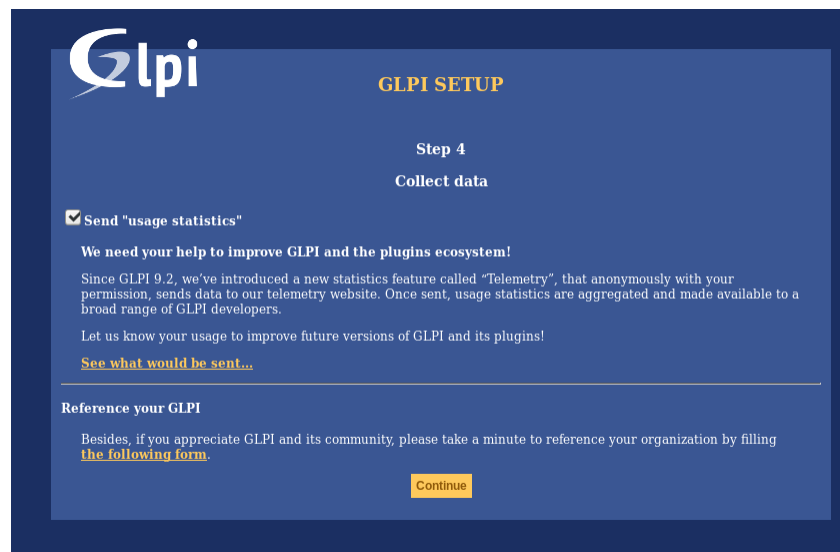
Tento krok inicializuje databázi s výchozími hodnotami.



V případě jakékoli chyby věnujte pozornost zobrazeným informacím.

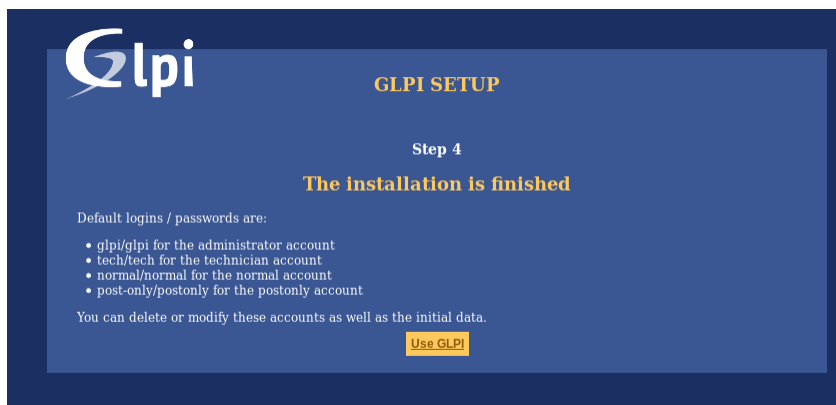
3.3.5 Telemetrické údaje

GLPI vás požádá o sdílení některých telemetrických údajů a registraci. Ani jedno není povinné.



3.3.6 Závěr instalace

V tomto kroku je představen souhrn instalace a vypsán seznam vytvořených uživatelů. Věnujte těmto informacím pozornost a pak kliknutím na Pokračovat přejděte do aplikace.



Poznámka: Výchozí uživatelské účty jsou:

- *glpi/glpi* účet správce,
 - *tech/tech* účet technika,
 - *normal/normal* účet běžného uživatele,
 - *post-only/postonly* účet pouze pro zadávání požadavků.
-

Varování: Ze zřejmých bezpečnostních důvodů byste měli tyto účty změnit (heslo) nebo smazat.

Před případným odebráním účtu *glpi*, nezapomeňte vytvořit jiný účet, který má profil *super-admin*.



In order to get timezones working on a MariaDB/MySQL instance, you will have to initialize Timezones data and grant GLPI database user read ACL on their table.

Varování: Enabling timezone support on your MySQL instance may affect other database in the same instance; be carefull!

Varování: Currently, MySQL and MariaDB have a maximum date limited to 2038-01-19 on fields relying on timestamp type!

4.1 Non windows users

On most systems, you'll have to initialize timezones data from your system's timezones:

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

You may want to check [MariaDB documentation about mysql_tzinfo_to_sql](#) and your system documentation to know where data are stored (if not in `/usr/share/zoneinfo`).

Do not forget to restart the database server once command is successfull.

4.2 Windows users

Windows does not provide timezones informations, you'll have to download and initialize data yourself.

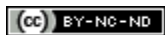
See [MariaDB documentation](#) about timezones.

4.3 Udělit přístup

Varování: Be carefull not to give your GLPI database user too large access. System tables should **never** grant access to app users.

In order to list possible timezones, your GLPI database user must have read access on `mysql.time_zone_name` table. Assuming your user is `glpi@localhost`, you should run something like:

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



Aktualizace

Poznámka: Jako při každé aktualizaci, je třeba před přechodem na novější verzi zazálohovat některá data:

- **zazálohujte svou databázi;**
 - 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*.

Varování: 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.

Varování: Neměli byste se pokoušet obnovit zálohu databáze do neprázdné databáze (například databáze, která byla z nějakého důvodu převedena na novější jen z části).

Před zahájením obnovení ze zálohy ověřte, že je cílová databáze prázdná a pak teprve se pokuste o aktualizaci. V případě nezdaru je třeba opět začít od obnovy do prázdné databáze.

Poznámka: Update process will automatically disable your plugins.

Poznámka: 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.



Nástroje pro příkazový řádek

Od GLPI verze 9.2.2 jsou nástroje pro příkazový řádek poskytovány jako podporované skripty a jsou k dispozici ve složce `scripts` (skripty) v archivu. V předchozích verzích se tyto skripty nacházely ve složce `tools` (nástroje) která není oficiální a proto není přítomná v instalačním archivu vydání.

Since GLPI 9.4.0, command line tools are being centralized in a console application (`bin/console`). Calling `php bin/console` from GLPI directory displays the list of available commands.

Poznámka: Pokud je v systému nainstalované APCu PHP rozšíření, nemusí se podařit ho použít z příkazového řádku, protože ve výchozím nastavení je tato možnost vypnutá. Změnit je to možné nastavením volby `apc.enable_cli` na `on` v souboru s nastaveními APCu.

Varování: When using cli tools, please check the system user you are currently logged in with, and permissions on files and directories. With a wrong user, logs, cache and other files may be created with rights that would not allow your webserver to read or write on those files!

6.1 Console options

For every console command, following options are available:

- `--config-dir=CONFIG-DIR` path of configuration directory to use, relative to current working directory (required only if a custom path is used)
- `-h, --help` zobrazí nápovědu k příkazu
- `--lang=LANG` output language code (default value is existing GLPI „language“ configuration or „en_GB“)
- `-n, --no-interaction` disable command interactive questions
- `--no-plugins` disable GLPI plugins during command execution
- `-q, --quiet` vypne výstup z příkazu

- `-v|vv|vvv`, `--verbose=VERBOSE` verbosity level: 1 for normal output, 2 for more verbose output and 3 for debug

6.2 Additional install and update tools

6.2.1 Kontrola požadavků

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 Zapnout/Vypnout režim údržby

GLPI provides a maintenance mode that can be activated prior to an update, and deactivated after all has been checked. Just use the `glpi:maintenance:enable` and `glpi:maintenance:disable` commands.

6.3 Instalace

The `php bin/console db:install` has been made to install GLPI database in CLI mode.

Možné volby pro tento příkaz jsou:

- `-r`, `--reconfigure` to enable overriding of any existing DB configuration file
- `-f`, `--force` to force execution of installation even if database is not empty
- `-L`, `--default-language=DEFAULT_LANGUAGE` default language of GLPI (*en_GB* per default)
- `-H`, `--db-host=DB_HOST` host name (*localhost* per default)
- `-P`, `--db-port=DB_PORT` database port (default MySQL port if option is not defined)
- `-d``, `--db-name=DB_NAME` jméno databáze
- `-u`, `--db-user=DB_USER` jméno uživatele databáze
- `-p`, `--db-password=DB_PASSWORD` database user's password (use it without value to be prompted for password)

If mandatory options are not specified in the command call, the console will ask for them.

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

See also *console options*.

6.4 Nastavení připojení k databázi

Nové ve verzi 9.5.0.

The `php bin/console db:configure` has been made to define database connection parameters in CLI mode.

Možné volby pro tento příkaz jsou:

- `-r`, `--reconfigure` to enable overriding of any existing DB configuration file
- `-H`, `--db-host=DB_HOST` host name (*localhost* per default)
- `-P`, `--db-port=DB_PORT` database port (default MySQL port if option is not defined)

- `-d`, --db-name=DB_NAME` jméno databáze
- `-u, --db-user=DB_USER` jméno uživatele databáze
- `-p, --db-password=DB_PASSWORD` database user's password (use it without value to be prompted for password)

If mandatory options are not specified in the command call, the console will ask for them.

See also *console options*.

6.5 Aktualizace

The `php bin/console db:update` has been made to update GLPI database in CLI mode from a previously installed version.

There is no required arguments, just run the command so it updates your database automatically.

Varování: Před jakýmkoli pokusem o aktualizaci nezapomeňte zazálohovat databázi!

Varování: 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`.

Možné volby pro tento příkaz jsou:

- `-u, --allow-unstable` allow update to an unstable version (use it with cautions)
- `-f, --force` force execution of update from v-1 version of GLPI even if schema did not changed
- `-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.glpi-project.org>)
- `--no-telemetry` disallow usage statistics sending to Telemetry service (<https://telemetry.glpi-project.org>)

See also *console options*.

6.6 Bezpečnostní klíč

Nové ve verzi 9.4.6.

Poznámka: GLPI key file is available for GLPI $\geq 9.4.6$ but is not mandatory. As of GLPI 9.5, using the key file will be mandatory.

In order to store some sensitive data, GLPI relies on a homemade encryption/decryption tool, which uses a key to:

- encrypt data before storing them in the database,
- decrypt data that has been retrieved from the database.

The `php bin/console glpi:security:change_key` command allows to change the key, if it has been compromised for example. By default, command will:

- generate a new key and store it in the key file,

- update all configured fields (for core and compatible plugins) to use the new key,
- update all configuration entries listed (for core and compatible plugins) to use the new key.

6.7 Various tools

6.7.1 Kontrola databázového schématu

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

Možné volby pro tento příkaz jsou:

- `--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.

If you have any diff, output will looks like :

```
$ 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 LDAP synchronizace

The bin/console `glpi:ldap:synchronize_users` command can be used to synchronize users against LDAP server informations.

Možné volby pro tento příkaz jsou:

- `-c, --only-create-new` only create new users
- `-u, --only-update-existing` only update existing users
- `-s, --ldap-server-id[=LDAP-SERVER-ID]` synchronize only users attached to this LDAP server (multiple values allowed)
- `-f, --ldap-filter[=LDAP-FILTER]` filter to apply on LDAP search
- `--begin-date[=BEGIN-DATE]` begin date to apply in „modifyTimestamp“ filter
- `--end-date[=END-DATE]` end date to apply in „modifyTimestamp“ filter
- `-d, --deleted-user-strategy[=DELETED-USER-STRATEGY]` force strategy used for deleted users:
 - 0: Preserve
 - 1: Put in trashbin
 - 2: Withdraw dynamic authorizations and groups
 - 3: Disable
 - 4: Disable + Withdraw dynamic authorizations and groups

See <http://php.net/manual/en/datetime.formats.php> for supported date formats in `--begin-date` and `--end-date` options.

See also *console options*.

6.7.3 Task unlock

The php bin/console `task:unlock` command can be used to unlock stucked cron tasks.

Varování: Keep in mind that no task should be stucked except in case of a bug or a system failure (database failure during cron execution for example).

Možné volby pro tento příkaz jsou:

- `-a, --all` unlock all tasks
- `-c, --cycle[=CYCLE]` execution time (in cycles) from which the task is considered as stuck (delay = task frequency * cycle)
- `-d, --delay[=DELAY]` execution time (in seconds) from which the task is considered as stuck (default: 1800)
- `-t, --task[=TASK]` itemtype::name of task to unlock (e.g: MailCollector::mailgate)

See also *console options*.

6.8 Plugins tools

Nové ve verzi 9.5.

Some command line tools are also available to manage plugins from command line:

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

In order to install MyGreatPlugin; you should end with something like:

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

Each of those plugin commands can take a plugin name as argument, or the `--all` flag to be ran on all plugins.

6.9 Migration tools

6.9.1 From MyISAM to InnoDB

Nové ve verzi 9.3.0.

Since version 9.3.0, GLPI uses the InnoDB engine instead of previously used MyISAM engine.

The `php bin/console glpi:migration:myisam_to_innodb` command can be used to migrate existing tables to InnoDB engine.

6.9.2 Missing timestamps builder

Nové ve verzi 9.1.0.

Prior to GLPI 9.1.0, fields corresponding to creation and modification dates were not existing.

The `php bin/console glpi:migration:build_missing_timestamps` command can be used to rebuild missing values using available logs.

6.9.3 Use timestamp data type

Nové ve verzi 9.5.0.

Many date fields were using the DATETIME type, but this does not allow to rely on timezones. Timezone support requires all fields to use TIMESTAMP data type, but this query can be very long and therefore is not included in the standard update process.

Using the `glpi:migration:timestamps` command will change those fields to the correct data type, but read *documentation on timezones* before.

Varování: Ensure to backup your database before!

6.9.4 Migrate Domains plugin

Nové ve verzi 9.5.0.

Domains in GLPI have evolved from a simple dropdown to a more complex object, including records management among others. Therefore, the Domains plugins feature are now included in core.

To migrate your plugin data; use the `glpi:migration:domains_plugin_to_core` command. Presence of the plugin is mandatory so checks can be run, you can use the `--without-plugin` switch but this is not recommended. If you were using an older version of the plugin than the one required, you can use the `--update-plugin` flag.

At the end, all domains types, domains and item relations will be migrated in core tables.

6.9.5 Migrate Racks plugin

Nové ve verzi 9.5.0.

Since GLPI 9.3.0, data center infrastructure management is available as a core feature. A migration script from Racks plugin was provided inside the `scripts` directory. Since GLPI 9.5.0, this migration script has been refactored and moved inside the CLI console.

To migrate your plugin data; use the `glpi:migration:racks_plugin_to_core` command. Presence of the plugin is mandatory so checks can be run, you can use the `--without-plugin` switch but this is not recommended. If you were using an older version of the plugin than the one required, you can use the `--update-plugin` flag.



7.1 SSL připojení k databázi

Nové ve verzi 9.5.0.

Po dokončení instalace můžete aktualizovat `` config / config_db.php`` a definovat parametry připojení SSL. Dostupné parametry odpovídají parametrům používaným programem *mysql* :: *ssl_set* () <<https://www.php.net/manual/en/mysqli.ssl-set.php>> _:

- `` \$dbssl`` definuje, zda by připojení mělo používat SSL (ve výchozím nastavení je na hodnotě `false` ,)
- `` \$dbsslkey`` název cesty k souboru klíče (ve výchozím nastavení není definováno *null*)
- `` \$dbsslcert`` název cesty k souboru certifikátu (ve výchozím nastavení není definován *null*)
- `` \$dbsslca`` název cesty k souboru certifikační autority (ve výchozím nastavení není definován *null*)
- `` \$dbsslcapath`` cesta k adresáři, který obsahuje důvěryhodné certifikáty SSL CA ve formátu PEM (ve výchozím nastavení není definována *null*)
- `$dbsslcacipher` seznam povolených šifer pro šifrování SSL (ve výchozím nastavení není definován *null*)

Varování: Prozatím není možné definovat parametry připojení SSL před nebo během procesu instalace. Nastavení parametrů SSL připojení musí být provedeno až po dokončení instalace.

