
GLPI Installation

GLPI Project, Teclib'

11 abr., 2023

1	Pré-requisitos	3
1.1	Servidor Web	3
1.1.1	Apache configuration	3
1.1.2	Nginx configuration	4
1.1.3	IIS configuration	5
1.2	PHP	5
1.2.1	Extensões obrigatórias	5
1.2.2	Extensões opcionais	6
1.2.3	Configuração	6
1.3	Banco de dados	6
2	Instalar GLPI	7
2.1	Escolha uma versão	7
2.2	Download	7
2.3	Instalação	8
2.4	Localizações de arquivos e diretórios	8
2.5	Pós-instalação	9
3	Assistente de Instalação	11
3.1	Selecione seu idioma	11
3.2	Licença	11
3.3	Instalar / Atualizar	12
3.3.1	Verificações de ambiente	13
3.3.2	Conexão de banco de dados	14
3.3.3	Escolha do banco de dados	14
3.3.4	Inicialização do banco de dados	15
3.3.5	Informação de telemetria	16
3.3.6	Fim da instalação	16
4	Fuso Horários	19
4.1	Utilizadores não Windows	19
4.2	Utilizadores Windows	19
4.3	Garantir o acesso	20
5	Atualizar	21
6	Ferramentas de linha de comando	23

6.1	Console options	23
6.2	Additional install and update tools	24
6.2.1	Check requirements	24
6.2.2	Enable/Disable maintenance	24
6.3	Instalar	24
6.4	Database connection configuration	24
6.5	Atualizar	25
6.6	Security key	25
6.7	Various tools	26
6.7.1	Database schema check	26
6.7.2	LDAP synchronization	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	Configuração avançada	31
7.1	Conexão SSL ao banco de dados	31

Esta documentação apresenta instruções de instalação do **GLPI**

GLPI (Gestion Libre de Parc Informatique) é uma solução de gerenciamento de ativos e helpdesk gratuita (como em «liberdade de expressão» e não como «cerveja de graça!»), acessível a partir de um navegador web, criado para gerenciar todos os problemas de gerenciamento de ativos, desde componentes de hardware e gerenciamento de inventários de software até o gerenciamento de helpdesk do usuário.

CAPÍTULO 1

Pré-requisitos

GLPI é um aplicativo da Web que precisará:

- um servidor web;
- PHP;
- um banco de dados.

1.1 Servidor Web

GLPI requer um servidor web que ofereça suporte a PHP, como:

- Apache 2 (ou mais recente);
- Nginx;
- Microsoft IIS.

1.1.1 Apache configuration

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

Aviso: 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),
```

(continues on next page)

(continuação da página anterior)

```

# you can use an Alias directive:
# Alias "/glpi" "/var/www/glpi/public"

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

    RewriteEngine On

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

```

Nota: 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.

Aviso: 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$ {
        # 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;
    }
}

```

(continues on next page)

(continuação da página anterior)

```
}
}
```

1.1.3 IIS configuration

Here is a configuration example for Microsoft IIS.

```
<?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>
```

1.2 PHP

Table 1: PHP Compatibility Matrix

GLPI Version	Minimum PHP	Maximum PHP
9.4.X	5.6	7.4
9.5.X	7.2	8.0
10.0.X	7.4	8.1

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

1.2.1 Extensões obrigatórias

As seguintes extensões PHP são necessárias para que o aplicativo funcione corretamente:

- `curl`: para autenticação CAS, verificação de versão do GLPI, Telemetria, etc;
- `fileinfo`: para obter informações extras dos arquivos;
- `gd`: para gerar imagens;
- `json`: para obter suporte para o formato de dados JSON;
- `mbstring`: gerenciar caracteres multibytes;
- `mysqli`: para conectar e consultar o banco de dados;

- `session`: para obter suporte a sessões de usuários;
- `zlib`: para efetuar backup e restaurar as funções da base de dados;
- `simplexml`;
- `xml`;
- `intl`.

1.2.2 Extensões opcionais

Nota: Mesmo que essas extensões não sejam obrigatórias, recomendamos que você as instale de qualquer maneira.

As seguintes extensões PHP são necessárias para alguns recursos extras do GLPI:

- `cli`: usar o PHP a partir da linha de comando (scripts, ações automáticas, etc.);
- `domxml`: usado para autenticação CAS;
- `ldap`: usar o diretório LDAP para autenticação;
- `openssl`: comunicações seguras;
- `xmlrpc`: usado para a API XMLRPC.
- `APCu`: pode ser usado para cache; entre outros (veja *configuração de caching (somente em francês)* <<http://glpi-user-documentation.readthedocs.io/fr/latest/advanced/cache.html>>)

1.2.3 Configuração

O arquivo de configuração do PHP (`php.ini`) pode ser adaptado para refletir as seguintes variáveis:

```
memory_limit = 64M ;           // max memory limit
file_uploads = on ;
max_execution_time = 600 ;     // not mandatory but recommended
session.auto_start = off ;
session.use_trans_sid = 0 ;    // not mandatory but recommended
```

1.3 Banco de dados

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

Para funcionar, o GLPI requer um servidor de banco de dados.



Proceda da seguinte forma:

1. *Configure your webserver*,
2. Escolha uma versão,
3. Baixe o arquivo,
4. Instale :)

2.1 Escolha uma versão

Nota: É altamente recomendado escolher a versão estável mais recente para um uso de produção.

O GLPI segue um esquema de versão semântico em 3 dígitos. O primeiro é o lançamento principal, o segundo o menor e o terceiro a versão fixa.

Os principais lançamentos podem vir com incompatibilidades importantes, bem como com novos recursos; As versões menores também podem trazer novos recursos, mas mantêm-se perfeitamente compatíveis dentro de uma versão importante.

As versões de correções só resolverão os problemas relatados sem adicionar nada novo.

2.2 Download

Aviso: No GitHub há sempre dois arquivos denominados *Source code* que não devem ser usados.

Vá para a seção *download* do [site GLPI](#) (ou obtenha o arquivo diretamente da versão [Github](#)) e escolha o arquivo `glpi-{version}.tgz`.

2.3 Instalação

A instalação GLPI em si é composta de três etapas:

1. Descompacte o arquivo em seu site;
2. Dê ao seu servidor web acesso de gravação aos diretórios `files` e `config`;
3. *Iniciar o assistente de instalação* (ou use o :ref. *script de instalação na linha de comando* `<cdline_install>`).

Uma vez que as três etapas foram concluídas o aplicativo está pronto para utilização.

Caso precise ativar uma configuração avançada, como parâmetros de conexão SSL, por favor consulte :doc:‘advanced configuration <advanced-configuration>’.

2.4 Localizações de arquivos e diretórios

Como muitos outros aplicativos web, o GLPI pode ser instalado apenas copiando o diretório inteiro para qualquer servidor web. No entanto, isso pode ser menos seguro.

Aviso: Todo arquivo acessível diretamente de um servidor web deve ser considerado inseguro!

O GLPI armazena alguns dados no diretório “`files`” e a configuração de acesso ao banco de dados é armazenada no diretório “`config`”. Mesmo que o GLPI forneça segurança para evitar que os arquivos sejam acessados diretamente de um servidor web, a melhor maneira é armazenar os dados fora da raiz da Web. Dessa forma, arquivos sensíveis não podem ser acessados diretamente do servidor web.

Existem algumas diretivas de configuração que você pode usar para conseguir isso (diretivas usadas nos pacotes downstream fornecidos):

- `GLPI_CONFIG_DIR`: defina o caminho para o diretório de configuração;
- “`GLPI_VAR_DIR`”: defina o caminho para o diretório “`files`”;
- “`GLPI_LOG_DIR`”: defina o caminho para os arquivos de log.

Nota: Existem muitas outras diretrizes de configuração disponíveis, as mencionadas anteriormente são as principais para se levar em conta para ter uma instalação mais segura.

A escolha de diretórios depende inteiramente de você, o seguinte exemplo segue as recomendações do [FHS](#).

O GLPI vai ser instalado no diretório `/var/www/glpi`, um host virtual específico na configuração do servidor web irá refletir esse caminho.

A configuração do GLPI será armazenada em `/etc/glpi`, basta copiar o conteúdo do diretório `config` para este local. O GLPI requer permissão de leitura neste diretório para funcionar; e permissão de gravação durante o processo de instalação.

Os dados do GLPI serão armazenados em `/var/lib/glpi`, apenas copie o conteúdo do diretório `files` para este local. O GLPI requer permissões de leitura e gravação neste diretório.

Os arquivos de log do GLPI serão armazenados em `/var/log/glpi`, não há nada para copiar aqui, basta criar o diretório. O GLPI requer acesso de leitura e gravação neste diretório.

Seguindo estas instruções, criaremos um arquivo `inc/downstream.php` no diretório do GLPI com os seguintes conteúdos:

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

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

Aviso: Os pacotes do GLPI certamente fornecerão um arquivo `inc/downstream.php`. Este arquivo não deve ser editado!

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.

Em seguida, crie um arquivo em `/etc/glpi/local_define.php` com os seguintes conteúdos:

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

Nota: Novo na versão 9.2.2.

Para as versões do GLPI anteriores à 9.2.2, a constante `GLPI_VAR_DIR` não existe. Era necessário definir todos os caminhos separadamente:

```
<?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 Pós-instalação

Uma vez instalado o GLPI, está quase terminado.

Uma etapa extra seria proteger (ou remover) o diretório de instalação. Como exemplo, você pode adicionar o seguinte à configuração do host virtual do Apache (ou no arquivo `glpi/install/.htaccess`):

```
<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>"
```

Neste exemplo, o acesso ao diretório *install* será limitado apenas ao localhost e exibirá uma mensagem de erro. Obviamente, você pode precisar adaptar isso às suas necessidades; consulte a documentação do seu servidor web.



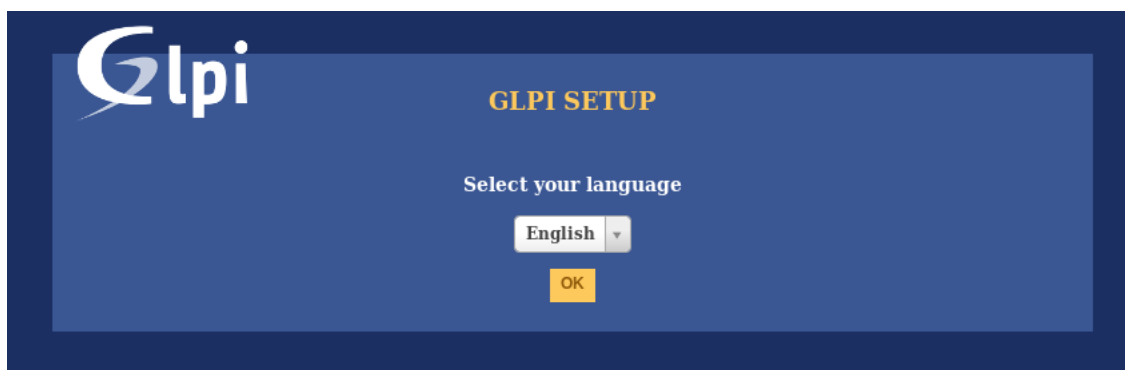
Assistente de Instalação

Para iniciar o processo de instalação, aponte seu navegador para o endereço principal do GLPI: `'https://{adresse_glpi}/<https://{adresse_glpi} />' _`

Quando o GLPI não está instalado; um processo passo-a-passo de instalação iniciará.

3.1 Selecione seu idioma

O primeiro passo permitirá que você escolha o idioma de instalação. Selecione seu idioma e clique em validar.



3.2 Licença

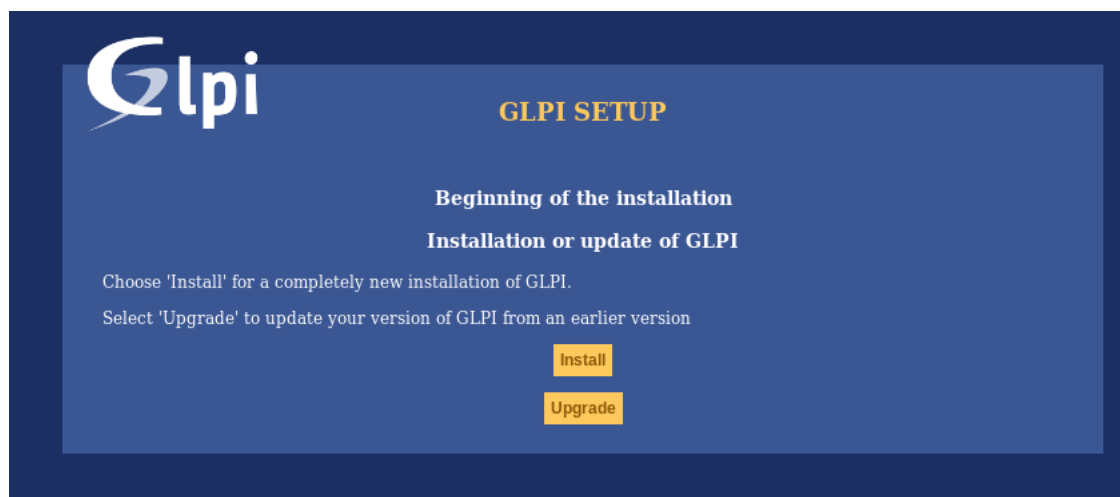
O uso do GLPI está sujeito à aprovação da licença GNU. Uma vez que os termos de licenciamento sejam lidos e aceitos, basta validar o formulário.



Se você não concordar com termos de licenciamento, não é possível continuar o processo de instalação.

3.3 Instalar / Atualizar

Esta tela permite escolher entre uma nova instalação GLPI ou uma atualização.



Clique em instalar.

3.3.1 Verificações de ambiente

Esta etapa verificará se os pré-requisitos foram atendidos. Caso contrário, não é possível continuar e uma mensagem de erro explícita informará o que está errado e o que fazer antes de tentar novamente.



The screenshot shows the GLPI SETUP interface at Step 0, titled "Checking of the compatibility of your environment with the execution of GLPI". It features a table with two columns: "Test done" and "Results". The table lists 34 tests, most of which passed with a green checkmark. Two tests failed: "ldap extension test" and "Zend OPcache extension test", both marked with a yellow warning icon and a message indicating they are not present. At the bottom, there is a question "Do you want to continue?" with two buttons: "Continue" and "Try again".

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	✓
xmlrpc 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

Alguns pré-requisitos são opcionais, será possível continuar o evento de instalação se eles não forem atendidos.

3.3.2 Conexão de banco de dados

Os parâmetros de conexão do banco de dados são solicitados.



The screenshot shows the GLPI Setup interface. At the top left is the GLPI logo. The title 'GLPI SETUP' is centered at the top. Below it, 'Step 1' and 'Database connection setup' are displayed. A white box titled 'Database connection parameters' contains three input fields: 'SQL server (MariaDB or MySQL)', 'SQL user', and 'SQL password'. An orange 'Continue' button is located at the bottom right of the setup area.

- *Servidor MySQL*: digite o caminho para o servidor MySQL, *localhost* ou *mysql.domain.tld* como exemplo;
- *Usuário MySQL*: digite o nome do usuário que está autorizado a se conectar ao banco de dados;
- *Senha MySQL*: digite a senha do usuário.

Uma vez que todos os campos estiverem devidamente preenchidos, valide o formulário.

Uma primeira conexão de banco de dados é então estabelecida. Se os parâmetros forem inválidos, uma mensagem de erro será exibida, e você terá que consertar os parâmetros e tentar novamente.

3.3.3 Escolha do banco de dados

Uma vez que a conexão com o servidor do banco de dados está OK, você deve criar ou escolher o banco de dados que deseja para o GLPI e iniciar ele.



Existem 2 maneiras de seguir:

- usar um banco de dados existente

Selecione este banco de dados na lista exibida. Validar para usar.

Aviso: O conteúdo selecionado do banco de dados será destruído na instalação.

- Criar um novo banco de dados

Escolha *Criar um novo banco de dados*, digite o nome do banco de dados no campo relevante e depois valide para criar a base.

Aviso: O usuário SQL deve ser capaz de criar um novo banco de dados para que esta opção funcione.

3.3.4 Inicialização do banco de dados

Esta etapa inicializa o banco de dados com valores padrões



Se houver algum erro, preste atenção nas informações exibidas.

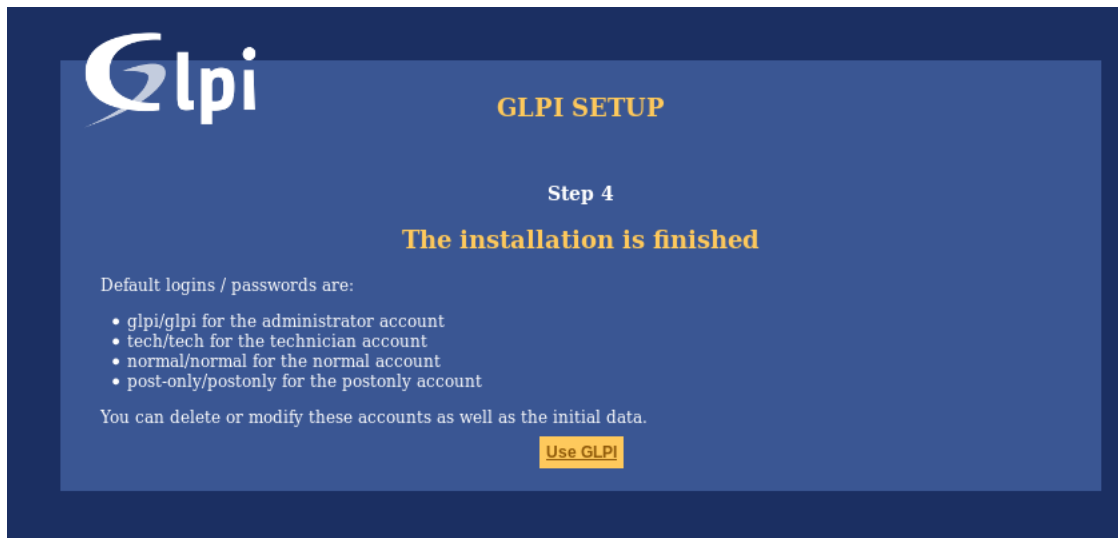
3.3.5 Informação de telemetria

O GLPI pedirá que você compartilhe algumas informações de telemetria e se registre. Isso não é obrigatório.



3.3.6 Fim da instalação

Esta etapa apresenta um resumo da instalação e fornece uma lista de usuários criados. Preste atenção a essas informações e valide para ir ao aplicativo.



Nota: As contas de usuário padrão são:

- *glpi/glpi* conta de administrador,
- *tech/tech* conta de técnico,
- *normal/normal* conta «normal»,
- *post-only/postonly* conta somente pós-publicação.

Aviso: Por razões de segurança, é aconselhável excluir ou editar essas contas.

Antes de remover a conta *glpi*, verifique se você criou outro usuário com o perfil *super-admin*.



Para que os fusos horários funcionem em uma instância do MariaDB/MySQL, você precisará inicializar os dados dos fusos horários e permitir que o usuário do banco de dados GLPI leia a ACL em sua tabela.

Aviso: Habilitar o suporte ao fuso horário na sua instância do MySQL pode afetar outro banco de dados na mesma instância; seja cuidadoso!

Aviso: Atualmente, o MySQL e o MariaDB têm uma data máxima limitada a 2038-01-19 em campos que dependem do tipo `timestamp`!

4.1 Utilizadores não Windows

Na maioria dos sistemas, você precisará inicializar os dados dos fusos horários a partir dos fusos horários do sistema:

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

Você pode verificar a documentação do [MariaDB sobre mysql_tzinfo_to_sql](#) e a documentação do sistema para saber onde os dados estão armazenados (se não estiver em `/usr/share/zoneinfo`).

Não esqueça de reiniciar o servidor de banco de dados quando o comando for bem-sucedido.

4.2 Utilizadores Windows

O Windows não fornece informações de fuso horário, você precisará baixar e inicializar os dados por conta própria.

Consulte a [documentação do MariaDB sobre fusos horários](#).

4.3 Garantir o acesso

Aviso: Cuidado para não conceder muito acesso ao usuário do banco de dados GLPI. As tabelas do sistema nunca devem conceder acesso aos usuários do aplicativo.

Para listar possíveis fusos horários, o usuário do banco de dados GLPI deve ter acesso de leitura na tabela “mysql.time_zone_name”. Supondo que seu usuário seja glpi@localhost, você deve executar algo como:

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



Atualizar

Nota: A cada processo de atualização, você deve fazer backup dos dados antes de iniciar qualquer atualização:

- ** backup do seu banco de dados **;
 - 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*.

Aviso: 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.

Aviso: Você não deve tentar restaurar um backup de banco de dados em um banco de dados não vazio (por exemplo, um banco de dados parcialmente migrado por qualquer motivo).

Verifique se o banco de dados está vazio antes de restaurar o backup e tente atualizar, e repita o processo caso ocorra alguma falha.

Nota: Update process will automatically disable your plugins.

Nota: 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.



Ferramentas de linha de comando

Desde o GLPI 9.2.2, as ferramentas de linha de comando são fornecidas como scripts suportados e estão disponíveis no diretório “ scripts “ do arquivo. Nas versões anteriores, esses scripts estavam presentes no diretório “ tools “ que não era oficial e, portanto, não estão no arquivo de lançamento.

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.

Nota: Se o APCu estiver instalado em seu sistema, ele poderá falhar na linha de comando já que a configuração padrão o desativa na linha de comando. Para mudar isso, configure “ `apc.enable_cli` “ para “ `on` “ no arquivo de configuração do APCu.

Aviso: 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 thos 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` displays command help
- `--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` disable command output

- `-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 Check requirements

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 Enable/Disable maintenance

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 Instalar

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

Possíveis opções para este comando são:

- `-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` database name
- `-u, --db-user=DB_USER` database user name
- `-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 Database connection configuration

Novo na versão 9.5.0.

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

Possíveis opções para este comando são:

- `-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` database name
- `-u, --db-user=DB_USER` database user name
- `-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 Atualizar

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.

Aviso: Não se esqueça de efetuar o backup do banco de dados antes de qualquer atualização!

Aviso: 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`.

Possíveis opções para este comando são:

- `-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 Security key

Novo na versão 9.4.6.

Nota: 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 Database schema check

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

Possíveis opções para este comando são:

- `--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 synchronization

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

Possíveis opções para este comando são:

- `-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.

Aviso: 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).

Possíveis opções para este comando são:

- `-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

Novo na versão 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

Novo na versão 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 exiting tables to InnoDB engine.

6.9.2 Missing timestamps builder

Novo na versão 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 re-build missing values using available logs.

6.9.3 Use timestamp data type

Novo na versão 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.

Aviso: Ensure to backup your database before!

6.9.4 Migrate Domains plugin

Novo na versão 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

Novo na versão 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 Conexão SSL ao banco de dados

Novo na versão 9.5.0.

Depois que a instalação estiver concluída, você poderá atualizar o `config/config_db.php` para definir os parâmetros de conexão SSL. Os parâmetros disponíveis correspondem aos parâmetros usados por `mysqli :: ssl_set ()`:

- `$dbssl` define se a conexão deve usar SSL (*false* por padrão)
- `$dbsslkey` nome do caminho para o arquivo de chave (*null* por padrão)
- `$dbsslcert` nome do caminho para o arquivo de certificado (*null* por padrão)
- `$dbsslca` Nome do caminho para o arquivo de autoridade de certificação (*null* por padrão)
- `$dbsslcapath` nome do caminho para um diretório que contém certificados CA SSL confiáveis no formato PEM (*null* por padrão)
- `$dbsslcipher` lista de cifras permitidas para criptografia SSL (*null* por padrão)

Aviso: Por enquanto, não é possível definir parâmetros de conexão SSL antes ou durante o processo de instalação. Isso deve ser feito assim que a instalação for concluída.

