c) Téléchargement et installation de GLPI :

c) Téléchargement et installation de GLPI :

Puis téléchargé le dossier compressé de GLPI 11.0.14 dans le dossier tmp (temporaire) :

```
cd /tmp

wget https://github.com/glpi-project/glpi/releases/download/11.0.4/glpi-11.0.4.tgz
```



Puis une fois téléchargé, le décompressé (l’argument /var/www/html c’est pour qu'il se mettent automatiquement dans le dossier des pages web d’apache) :

```
sudo tar -xzvf glpi-11.0.4.tgz -C /var/www/html
```

Puis changé le propriétaire du dossier par www-data pour permettre leur bon fonctionnement avec apache2 :

```
sudo chown www-data /var/www/html/glpi/ -R
```

GLPI-project recommande pour des question de récurité, de mettre en place ailleur du dossier glpi même, les fichiers de configuration, le dossier contenant la moitier des fichiers de GLPI et les logs.

Dans un premier temps, on va déplacé le dossier **config** de GLPI.

Nous allons mettre le dossier de configuration de GLPI dans "**/etc/glpi/**", donc création du fichier puis affectation du propriétaire et finir par le déplacement :

```
sudo mkdir /etc/glpi
sudo chown www-data /etc/glpi/
sudo mv /var/www/html/glpi/config /etc/glpi
```

```
root@GLPI-VM:/tmp# sudo mkdir /etc/glpi
sudo chown www-data /etc/glpi/
sudo mv /var/www/html/glpi/files /var/lib/glpi
```

Maintenant, passons au dossier **lib**, ce qui contiens par exemple les **plugins,** des **CSS**, etc. On fais également comme au dessus :

```
sudo mkdir /var/lib/glpi
sudo chown www-data /var/lib/glpi/
sudo mv /var/www/html/glpi/files /var/lib/glpi
```

```
root@GLPI-VM:/tmp# sudo mkdir /var/lib/glpi
sudo chown www-data /var/lib/glpi/
sudo mv /var/www/html/glpi/files /var/lib/glpi
```

Finissons par les **logs** mais comme nous venons d'installé GLPI, nous allons rien déplacé, juste crée et changé le propriétaire du dossier :

```
sudo mkdir /var/log/glpi
sudo chown www-data /var/log/glpi
```

Dans le second temps, on va créer des fichier **PHP** dans GLPI pour changer les liens des différents dossiers que nous avons modifiés précédemment.

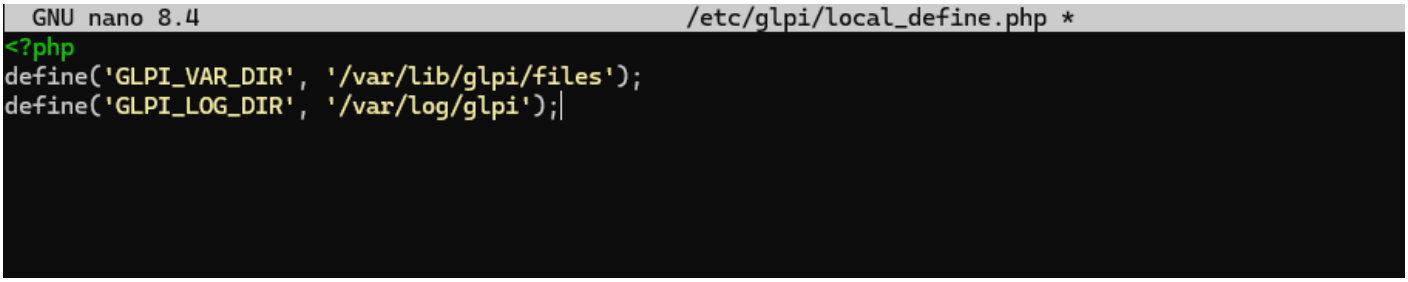
On va en premier crée un premier fichier pour définir le dossier "**config**" et des **variable** qui nous servira dans un second temps :

```
sudo nano /var/www/html/glpi/inc/downstream.php
```

```
 GNU nano 8.4                          /var/www/html/glpi/inc/downstream.php *
<?php
define('GLPI_CONFIG_DIR', '/etc/glpi/');
if (file_exists(GLPI_CONFIG_DIR . '/local_define.php')) {
    require_once GLPI_CONFIG_DIR . '/local_define.php';
}
```

Puis on va crée un second fichier php pour permettre de donnée des variables. Pour nous ici, se seras que pour les dossiers **files** et **log** mais on peux en mettre bien d'autre :

```
sudo nano /etc/glpi/local_define.php
```



Nous avons finit avec le déplacement de fichier, nous passons maintenant à la configuration web avec la création d'un fichier de configuration pour **apache2** :

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

```
<VirtualHost *:80>
    ServerName glpi.lerenard.eu
    DocumentRoot /var/www/html/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 NOT target the GLPI directory its>
    # Alias "/glpi" "/var/www/glpi/public"

    <Directory /var/www/html/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>
```

```
<VirtualHost *:80>
    ServerName glpi.lerenard.eu
    DocumentRoot /var/www/html/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 NOT target the GLPI directory its>
    # Alias "/glpi" "/var/www/glpi/public"

    <Directory /var/www/html/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>
```

Au début il y à "**ServerName**", vous pouvez mettre ce que vous voulez, mais c'est recommandé de mettre le même nom de domaine qu'il y à en interne (si vous avez un nom de domaine en local), si vous désirez de mettre un certificat TLS/SSL pour l'https par exemple.

Configuration de apache2 pour dès que vous tapé l’ip sur un navigateur, il fasse la redirection vers GLPI et activation de PHP8.4-fpm pour l'exécution plus optimisé de script (et comme on en a fait au dessus) :

```
sudo a2ensite glpi.conf #(permet d’activé le fichier conf créé au-dessus pour glpi sur apache)
sudo a2dissite 000-default.conf #(permet de désactivé le fichier 000-default.conf qui fais référence à index.html)
sudo a2enmod rewrite #(permet l’activation du module rewrite pour apache2)
sudo a2enmod proxy_fcgi setenvif
sudo a2enconf php8.4-fpm
sudo systemctl restart apache2 #(redémarrage d’apache2)
```

Revision #12
Created 2025-12-22 21:15:40 UTC by Renard
Updated 2025-12-23 23:14:59 UTC by Renard