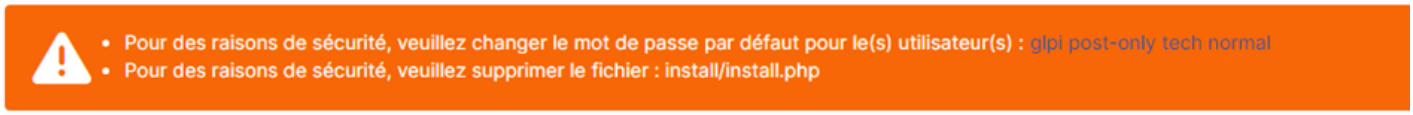
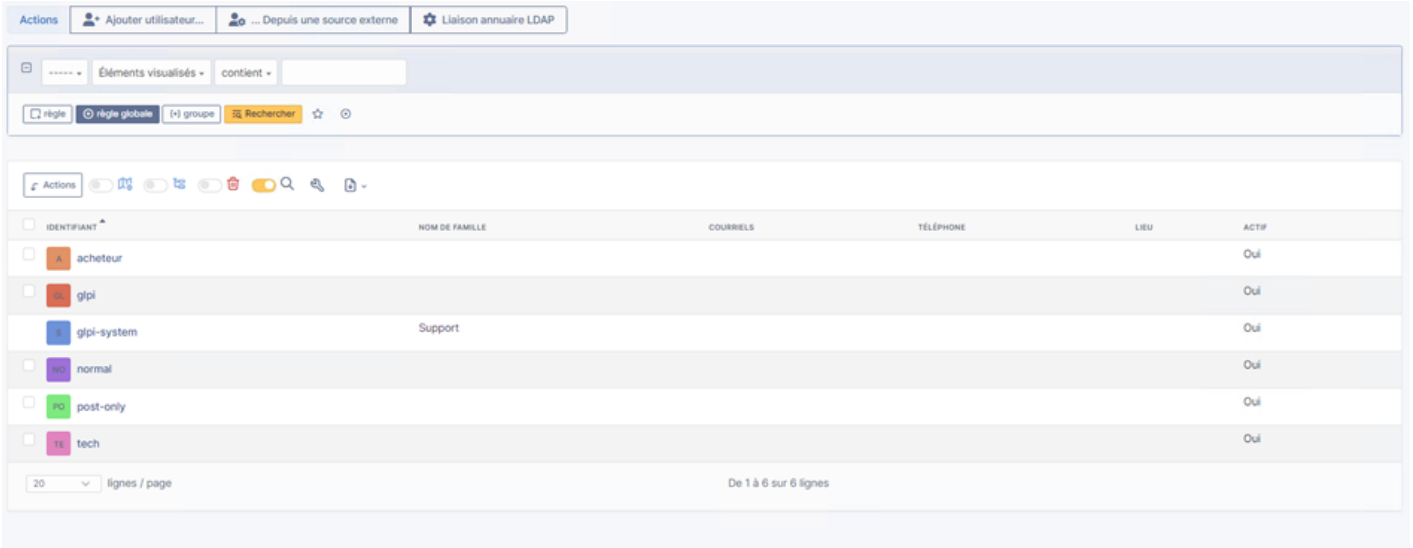
e) Sécurisation de GLPI

e) Sécurisation de GLPI :

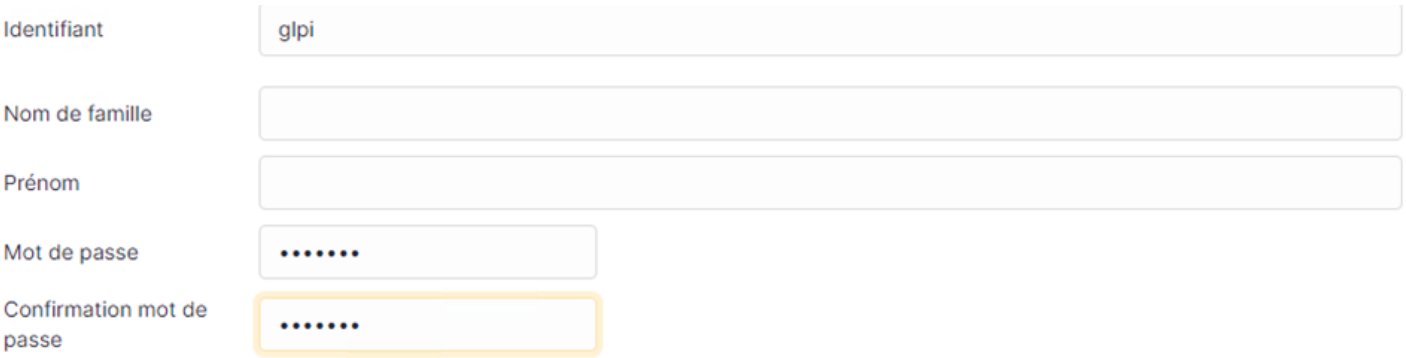
Pour des questions de sécurité nous allons voir comment changé.



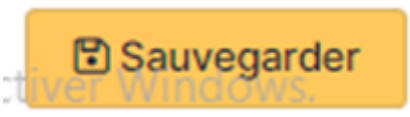
Aller dans Administration>Utilisateurs :



Choisissez un compte ou vous souhaité changé le mot de passe :



Puis faite sauvegarder.



Vous devriez plus voir le message :

Également, on doit supprimé aussi install.php, cela évitera tous problème si une personne relance une réinstallation. Il se trouve dans /var/www/html/glpi/install.

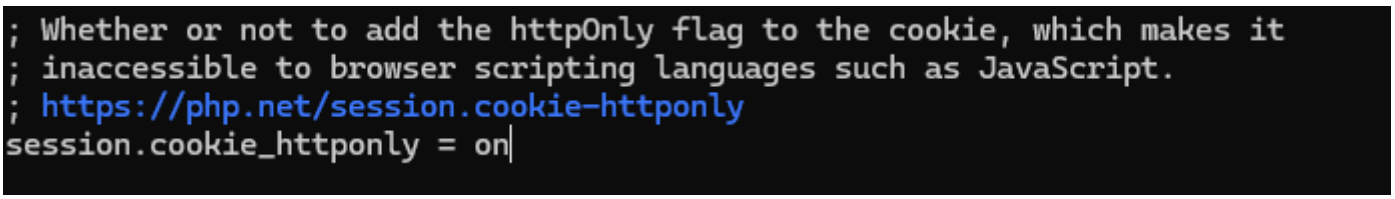
Faite rm install.php :

Dans un second temps, on va sécurisé php-fpm que l'on a activé précédament. La documentations GLPI recommande cela.

On doit sécurisé les cookie, on va donc allé ouvrir le fichier php.ini dans /etc/php/8.4/fpm/ :

```
sudo nano /etc/php/8.4/fpm/php.ini
```

Nous cherchons "**session.cookie_httponly**" pour le mettre sur "**on**" :



On viens d'activé la sécurité de base pour les cookies, maintenant nous allons configuré la fasson dont sont géré les cookie pour évité certaines attaques comme CSRF (Cross-Site Resquest Forgery).

Nous cherchons donc maintenant "**session.cookie_samesite"** pour le mettre sur la valeur "**Lax**" :

```
; Add SameSite attribute to cookie to help mitigate Cross-Site Request Forgery (CSRF/XSRF)
; Current valid values are "Strict", "Lax" or "None". When using "None",
; make sure to include the quotes, as `none` is interpreted like `false` in ini files.
; https://tools.ietf.org/html/draft-west-first-party-cookies-07
session.cookie_samesite = Lax
```

Pour finir, on doit édité notre fichier de configuration apache2 pour le forcé apache2 à utilisé php-fmp :

```
 GNU nano 8.4                                  /etc/apache2/sites-enabled/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 itself.
    # 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>
 <FilesMatch \.php$>
    SetHandler "proxy:unix:/run/php/php8.4-fpm.sock|fcgi://localhost/"
 </FilesMatch>
</VirtualHost>
```

Puis redémarrons :

```
sudo systemctl restart apache2
```

```
root@GLPI-VM:/home/glpi# sudo systemctl restart apache2
root@GLPI-VM:/home/glpi#
```

Voilà la fin de la sécurisation de GLPI. Mais cela devrais évolué avec la mise en place d'https dans le futur.

Revision #3
Created 2025-12-22 21:15:42 UTC by Renard
Updated 2025-12-23 23:11:41 UTC by Renard