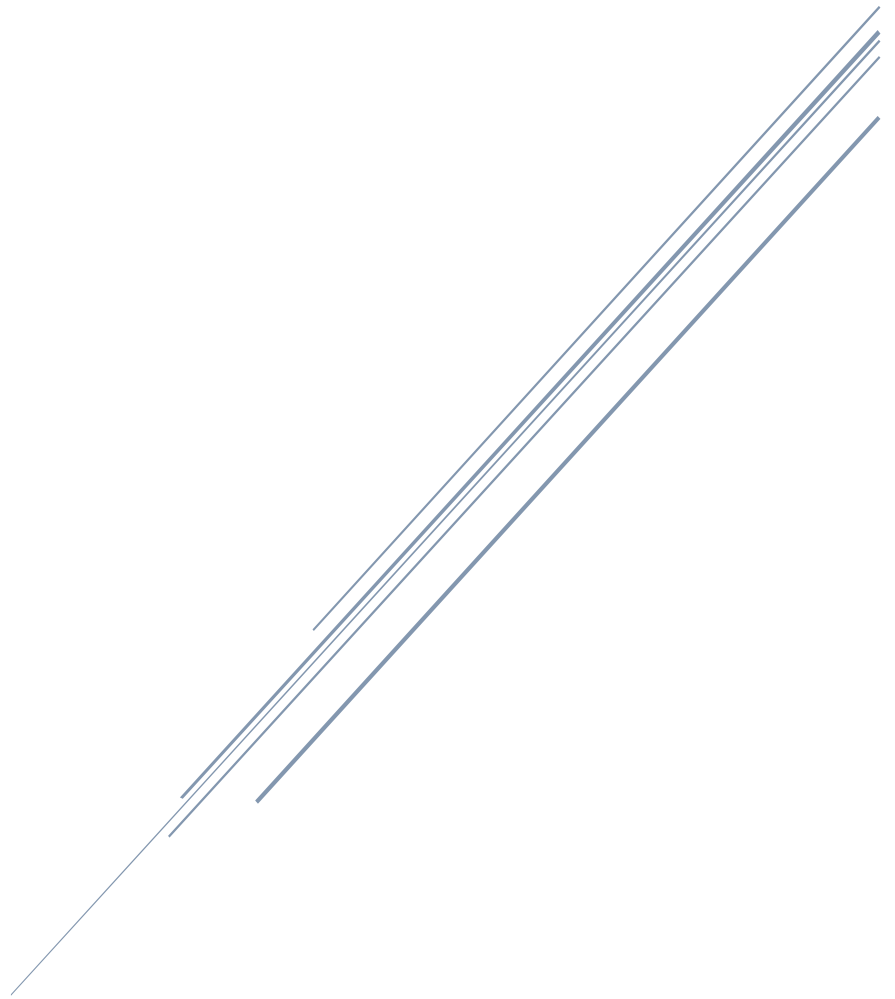


DESCRIPTION D'UNE MISSION

BTS SIO SISR



Sommaire:

Le cahier des charges3-4

Contexte	3
Expressions des besoins	3
Outils disponibles	3
Description de l'existant.....	4
Délais	4

Mise en oeuvre..... 5-19

Le choix de GLPI	5
Installation debian 11	6-9
Installation des dépendances de GLPI.....	10-13
Installation de GLPI.....	14-18
Découverte des différentes fonctions	19

Bilan.20

Le cahier des charges

Contexte :

L'entreprise Madnot vient de faire l'acquisition de nouveau bureau situé en région parisienne, une nouvelle infrastructure systèmes et réseau a été précédemment mise en place. L'entreprise aimerait avoir une gestion du nouveau parc informatique, nous avons donc en tant qu'administrateur systèmes et réseau reçus la mission de déployer un logiciel de gestion de parc informatique.

Expression des besoins :

Madnot souhaite le déploiement d'un logiciel permettant la gestion de leur parc informatique :

- Inventaire de toutes les ressources informatiques de la société.
- Systèmes de ticketing pour les salariés.
- Gestions et suivie des versions des logiciels

Outils disponibles :

Un serveur nous a été mis à disposition, il s'agit d'un HP BL460C G8 (Blade) dans un HP C7000. Nous disposons d'un accès IPMI (ILO) à ce serveur. Un hyperviseur ESXI est installé sur celui-ci.



Ci-dessus à gauche le serveur HP BL460C G8 mis à disposition, il s'agit d'une lame qui s'insère dans un châssis ici un HP C7000 (à droite) qui comporte 16 lames.




Description de l'existant :

L'entreprise dispose pour le moment de 2 ordinateurs, il dispose d'un routeur fourni par orange dans le cadre d'une FTO. Un HP C7000 est présent dans la baie réseau de l'entreprise, une lame HP BL460C G8 déjà installer sous ESXI nous ai mis à disposition pour nos besoins.

Délais :

Nous disposons d'une journée entière pour déployer le logiciel de gestion de parc informatique (GLPI).

Voici ci-dessous un diagramme de Gantt qui représente l'avancement de la mission :

	10h	14h	18h
Installation de GLPI			
Mise en œuvre			
Test des différentes fonction			

Mise en oeuvre :

Le choix de GLPI :

Pour répondre au besoin d'un logiciel de gestion de parc nous avons choisi GLPI car c'est un outil open source sous licence GPL (General Public License) V3 maintenu par Teclib, l'un des acteurs majeurs de la tech française. En cela, il est libre d'être installé, modifié et copié dans notre système d'information.

Les avantages sont donc qu'il est gratuit et français, GLPI est leader dans le domaine des logiciels de gestions et il est très simple d'utilisation.

Nos technicien et administrateur système et réseau ont bien plus de connaissance sur ce logiciel qui est couramment utilisé que sur d'autres logiciels de gestions de parc comme spiceworks.

C'est donc GLPI que nous allons déployer dans notre infrastructure.

GLPI nous permettra donc d'inventorier et de gérer le matériel informatique ainsi que le matériel réseau de l'entreprise.

Via GLPI nous pourrons aussi gérée les logiciels déployer pour nos utilisateurs et gérée leurs versions et donc leur maintien à jour pour assurer notamment le plus de sécurité possible sur notre infrastructure, car un logiciel qui n'est pas à jour est une potentielle faille de sécurité.

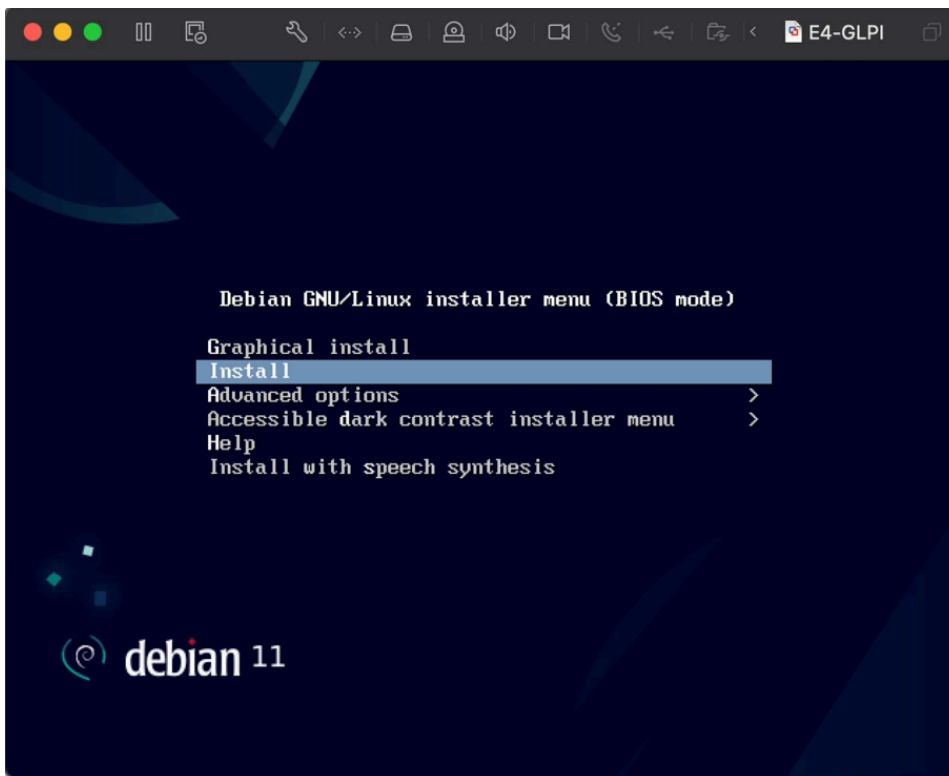
GLPI va aussi nous permettre de mettre en place un système de ticketing qui permettra à l'utilisateur de notre parc de faire remonter directement à notre service les éventuels problèmes informatiques qu'il pourrait rencontrer

Nous allons ici déployer uniquement GLPI, nous pourrons par la suite ajouter certains plugins notamment Fusion inventory qui est un logiciel servant à l'inventaire et la maintenance d'un parc. Cela permettra d'inventorier le matériel avec un agent ou en SNMP, l'utilisation du wake-on-lan, le déploiement de logiciels et la découverte du matériel réseau connecté via Netbios , NMAP et SNMP.

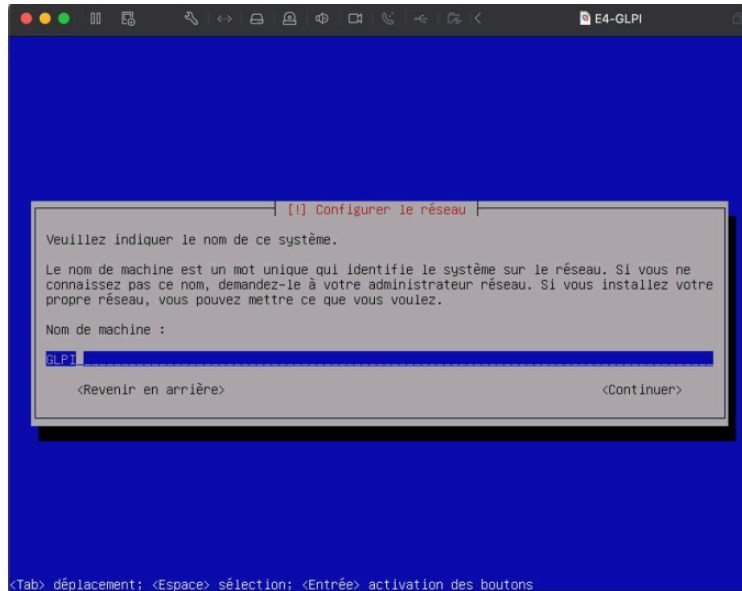
Installation du système d'exploitation :

Pour l'installation de notre serveur GLPI nous avons choisi le système d'exploitation Debian qui est un Linux composé exclusivement de logiciel libre.

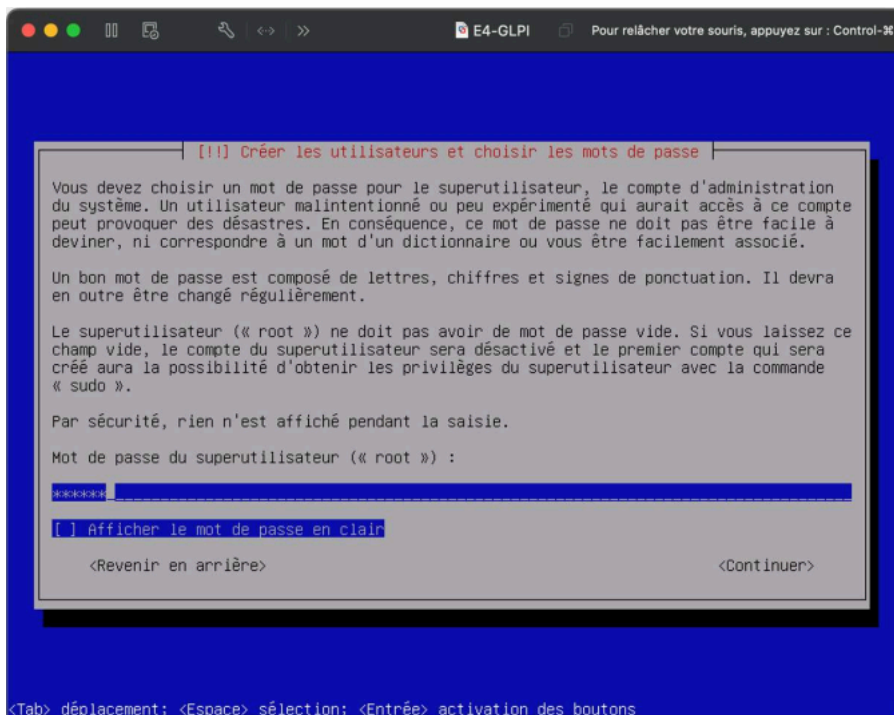
Après avoir sélectionné l'OS debian 11 durant la configuration de la machine virtuelle, nous pouvons la démarrer et suivre l'installation du système d'exploitation, nous sélectionnons une installation classique sans interface graphique :



Après avoir saisi les paramètres de langage ainsi que de localisation, nous entrons le nom de notre machine :

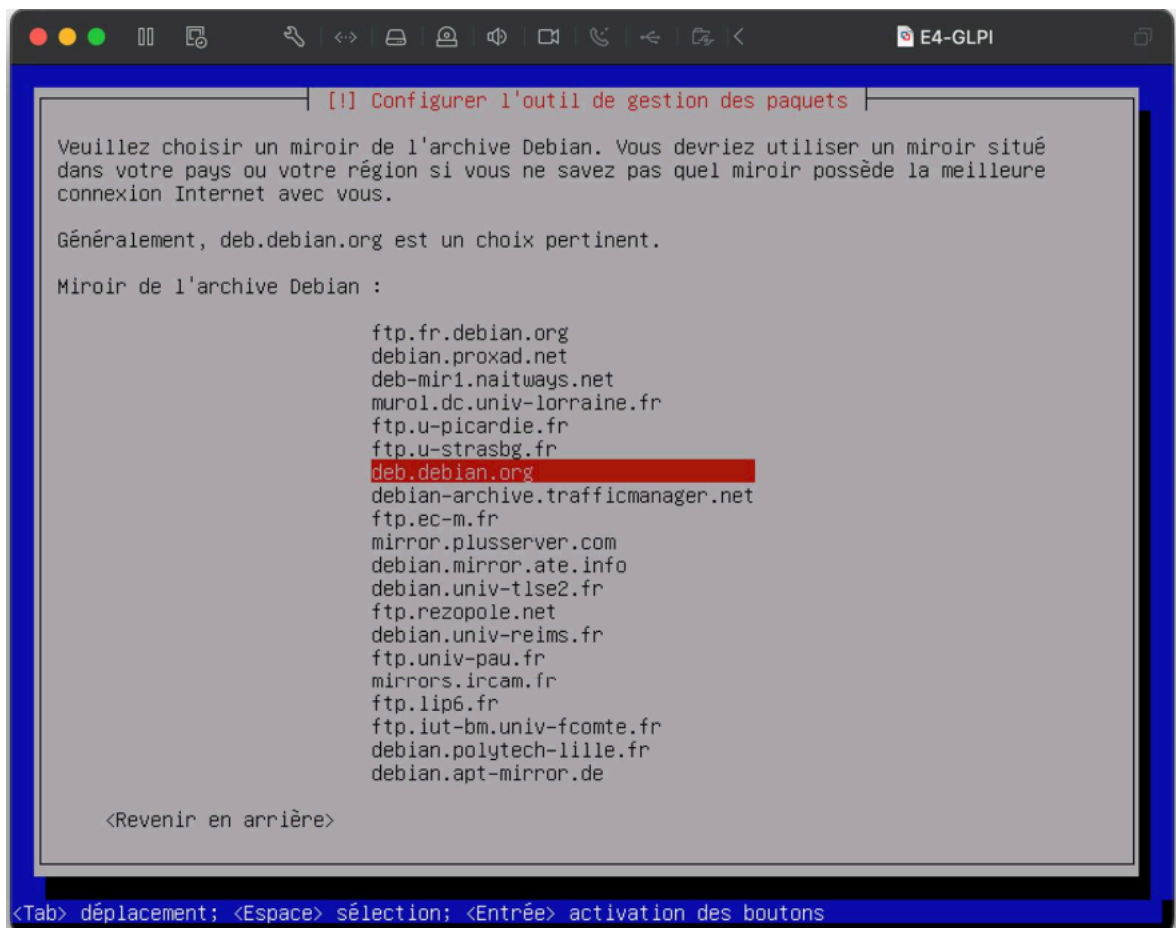


Nous entrons ensuite le mot de passe super utilisateur (root) :

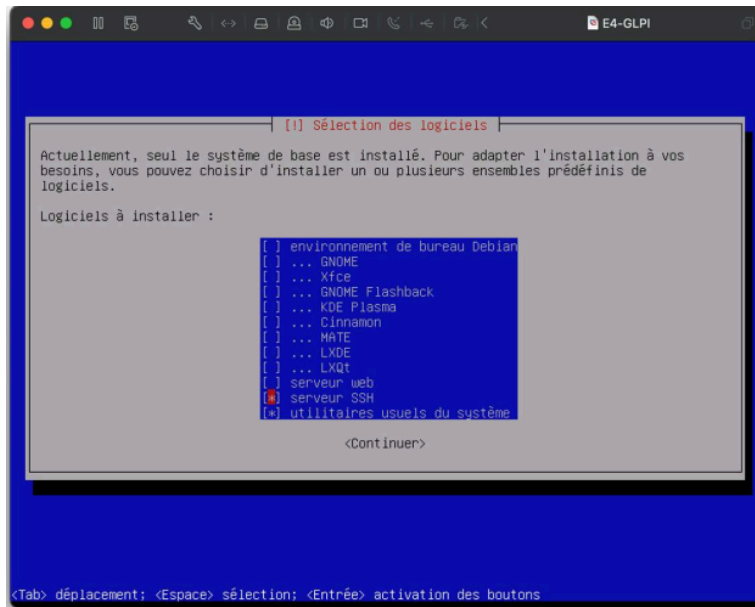


Nous sélectionnons ensuite un miroir Debian situé en France (cela améliore la vitesse de téléchargement), nous sélectionnons ici deb.debian.org

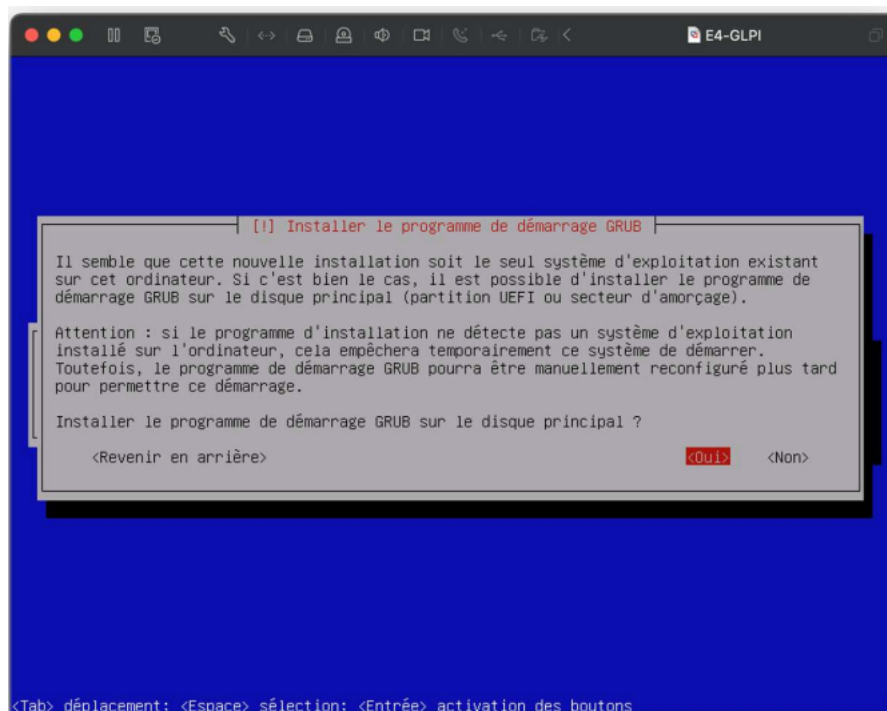
Un miroir d'archives est un référentiel de tous les fichiers de composants officiellement approuvés et testés d'une distribution Linux et de ses programmes associés à partir duquel un programme appelé gestionnaire de packages peut récupérer tous les fichiers et leurs dépendances (packages) requises pour installer cette distribution et tous les programmes dont l'utilisateur a besoin.



Nous n'utiliserons pas d'environnement de bureau, nous allons installer un serveur SSH qui nous sera utile pour accéder à distance au serveur.



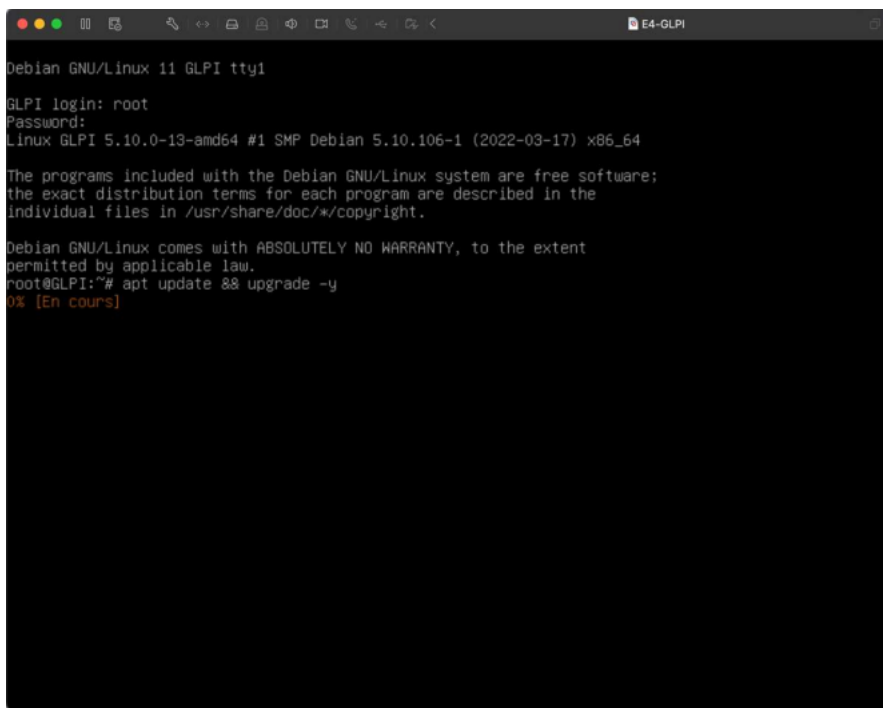
Nous installons le programme de démarrage GRUB sur le disque principal, la machine va ensuite redémarrer et l'installation du système d'exploitation sera alors terminée :



Installation des dépendances de GLPI :

GLPI nécessite plusieurs dépendances pour fonctionner, notamment apache (serveur web) et maria DB (Base de données)

Nous allons tout d'abord mettre à jour le serveur :



```
Debian GNU/Linux 11 GLPI tty1
GLPI login: root
Password:
Linux GLPI 5.10.0-13-amd64 #1 SMP Debian 5.10.106-1 (2022-03-17) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@GLPI:~# apt update && upgrade -y
0% [En cours]
```

Nous allons ensuite installer apache 2 (serveur web) :

- apt-get install apache2 php libapache2-mod-php

Nous installons ensuite PHP :

- apt-get install php-imap php-ldap php-curl php-xmlrpc php-gd php-mysql php-cas

Pour finir la dernière dépendance est mariaDB (Base de données) :

- apt-get install mariadb-server

Nous allons ensuite sécuriser l'accès à la base de données :

Cela permet de :

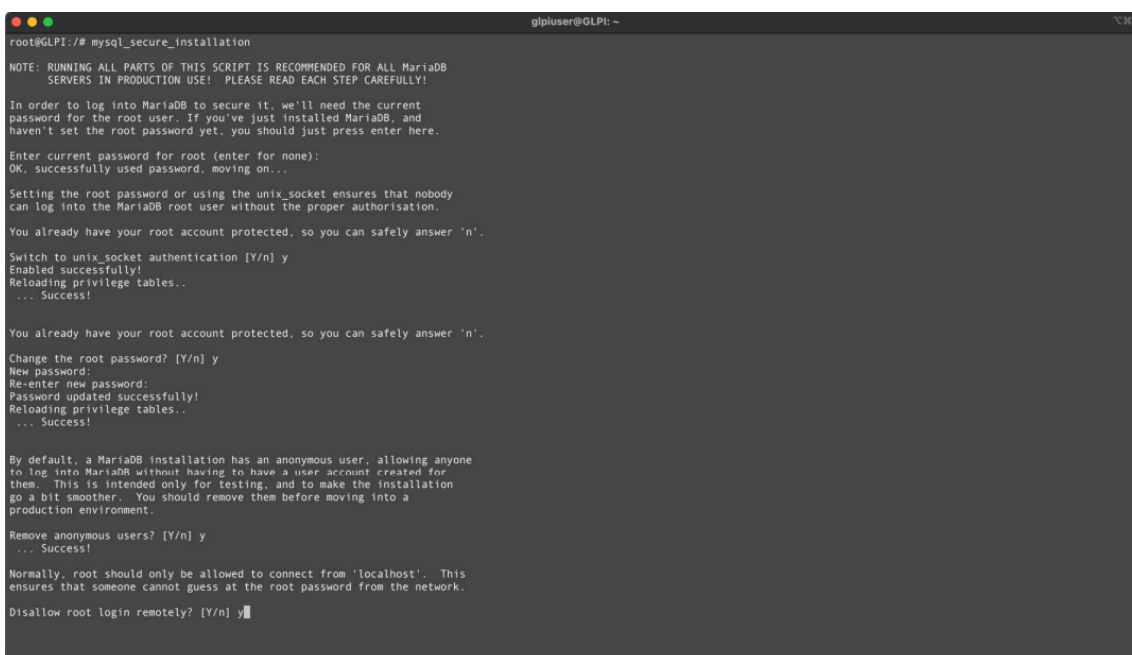
Définir un mot de passe pour les comptes root.

Supprimer les comptes root qui sont accessibles depuis l'extérieur de l'hôte local.

Supprimer les comptes d'utilisateurs anonymes.

Supprimer la base de données de test, à laquelle les utilisateurs anonymes peuvent accéder par défaut.

« mysql_secure_installation »



```
root@GLPI:~# mysql_secure_installation
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] y
Enabled successfully!
Reloading privilege tables..
... Success!

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] y
```

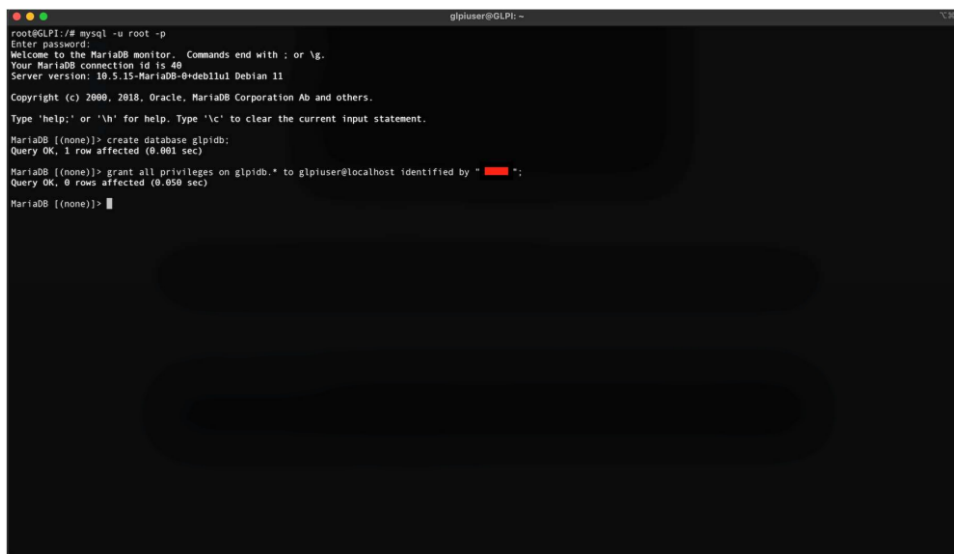
Nous téléchargeons les modules complémentaires nécessaires au bon fonctionnement de GLPI :

- apt-get install apcupsd php-apcu

Nous pouvons maintenant créer la base de données qui sera utilisée par GLPI :

- Mysql -u root -p (connexion root a mysql)
- Create glpidb (création de la base de données)
- grant all privileges on glpidb.* to glpiuser@localhost identified by "password";

(Ajout des privilèges à l'utilisateur « glpiuser » sur la base de données)



```
root@GLPI:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 48
Server version: 10.5.15-MariaDB-0+deb11u1 Debian 11

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database glpidb;
Query OK, 1 row affected (0.001 sec)

MariaDB [(none)]> grant all privileges on glpidb.* to glpiuser@localhost identified by " ";
Query OK, 0 rows affected (0.050 sec)

MariaDB [(none)]>
```

Nous procédons ensuite à l'installation de phpMyAdmin :

- apt-get install phpmyadmin

Nous sélectionnons apache2 pour lui indique le serveur web a utilisé.

Avant de procéder à l'installation de GLPI nous allons sécuriser l'accès au répertoire contenant GLPI en modifiant la configuration du site web par défaut d'apache

« nano /etc/apache2/sites-available/000-default.conf »

Nous allons ajouter sous la ligne « DocumentRoot »

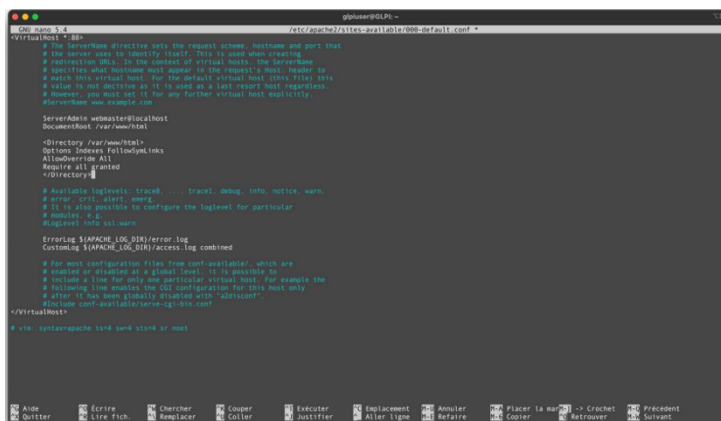
<Directory /var/www/html>

Options Indexes FollowSymLinks

AllowOverride All

Require all granted

</Directory>



```
glpiuser@GLPI: ~  
$ nano /etc/apache2/sites-available/000-default.conf  
#  
# The ServerName directive sets the request scheme, hostname and port that  
# the server uses to identify itself. This is used when creating  
# redirection URLs. In the context of virtual hosts, the ServerName  
# specifies what hostname must appear in the request's Host: header to  
# match this virtual host. For the default virtual host (this file) this  
# value is not required as it is covered by a last-virtual host requirement.  
# However, you must set it for any further virtual host explicitly.  
#  
#ServerName www.example.com  
ServerAdmin webmaster@localhost  
DocumentRoot /var/www/html  
  
<Directory /var/www/html>  
Options Indexes FollowSymLinks  
AllowOverride All  
Require all granted  
</Directory>  
  
# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,  
# error, crit, alert, emerg.  
# It is also possible to configure the loglevel for particular  
# modules, e.g.  
#LogLevel info ssl:warn  
  
ErrorLog ${APACHE_LOG_DIR}/error.log  
CustomLog ${APACHE_LOG_DIR}/access.log combined  
  
# For most configuration files from conf-available/, which are  
# enabled or disabled at a global level, it is possible to  
# include a line for only one particular virtual host. For example the  
# following line enables the CGI configuration for this host only  
# after it has been globally disabled with "CGIScript".  
#Include conf-available/cgis-script.conf  
  
</VirtualHost>  
  
# vim: syntax=apache ts=4 sw=4 sh=0
```

Nous pouvons ensuite redémarrer apache 2 pour appliquer les modifications :

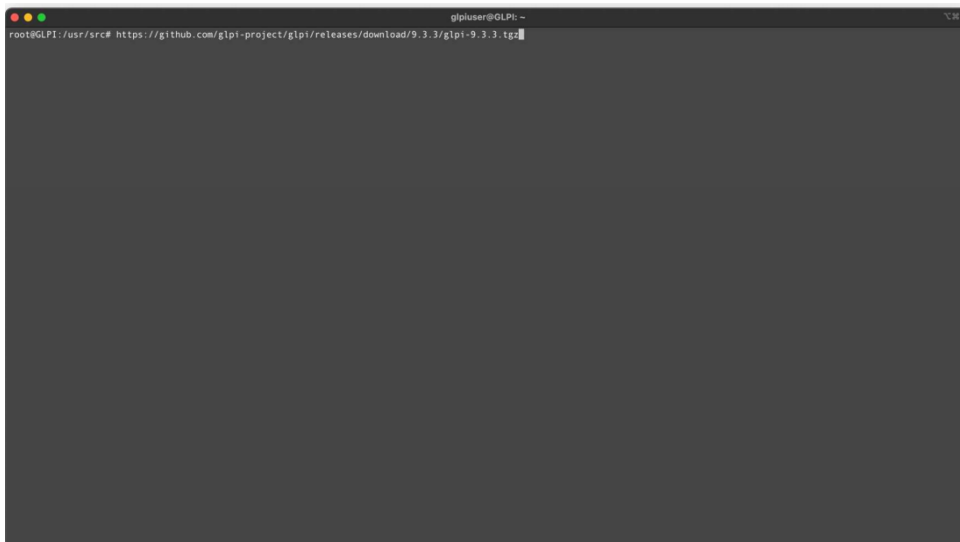
« systemctl restart apache2 »

Installation de GLPI :

Maintenant que toutes les dépendances de GLPI sont installées, nous pouvons procéder à son installation :

Nous nous rendons dans le dossier `/usr/src/` puis nous téléchargeons via `wget` l'archive de GLPI :

(<https://github.com/glp-project/glp/releases/download/9.3.3/glp-9.3.3.tgz>)



Nous décompressons ensuite l'archive dans le dossier `/var/www/html` qui est le dossier d'apache2 avec la commande suivante :

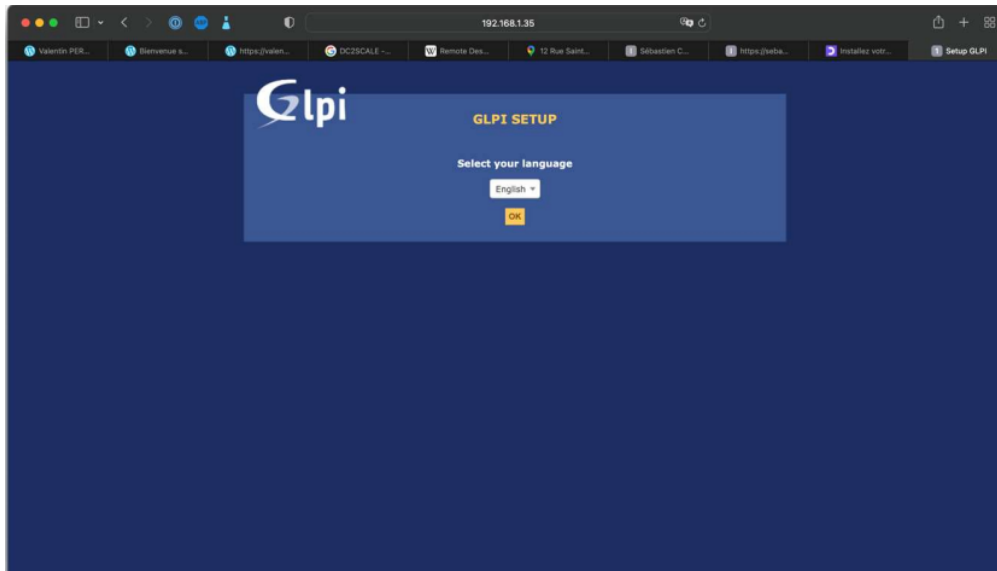
```
tar -xvzf glpi-9.3.3.tgz -C /var/www/html
```

Avant de pouvoir accéder à l'interface web de GLPI nous devons attribuer les droits au serveur LAMP d'agir sur le fichier disponible dans `/var/www/html/glp` pour cela nous utilisons la commande :

```
chown -R www-data /var/www/html/glp/
```

Nous pouvons maintenant accéder à l'interface web de GLPI via l'ip suivie de /glpi. L'ip de notre serveur est ici 192.168.1.35.

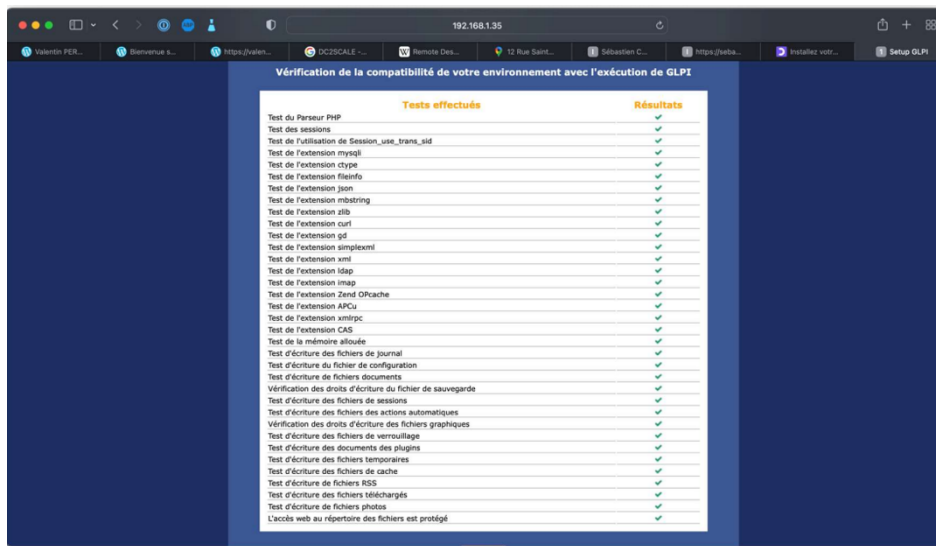
192.168.1.35/glpi



Nous suivons la configuration de l'interface web, langue, licence, puis nous sélectionnons Installer.



Nous retrouvons ensuite une Vérification de la compatibilité de notre environnement avec l'exécution de GLPI.



Nous configurons ensuite la connexion à la base de données (glpiid) :



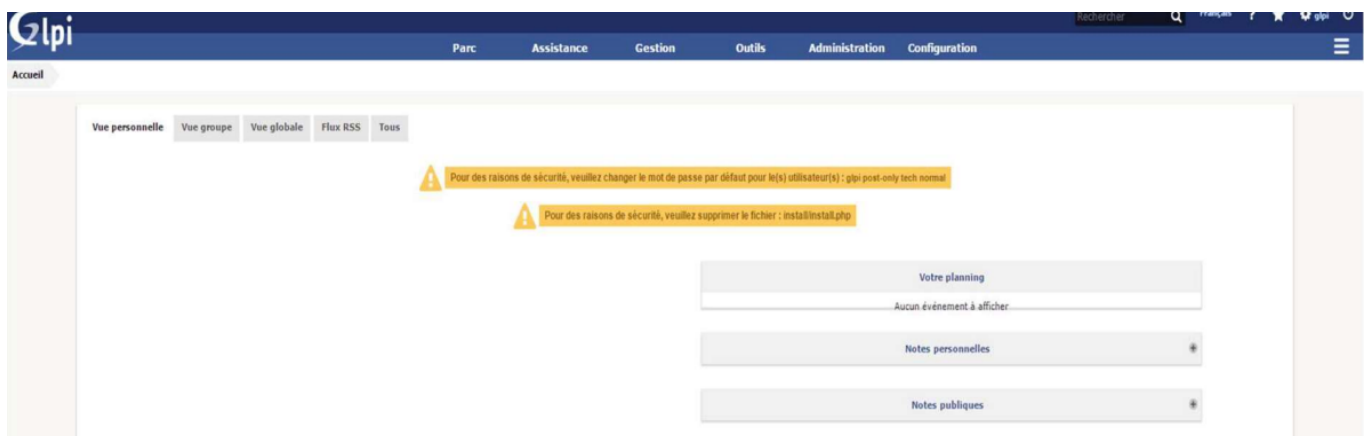
Nous pouvons maintenant nous connecter à l'interface web de GLPI, les identifiants par défaut sont

Username : glpi

Password : glpi



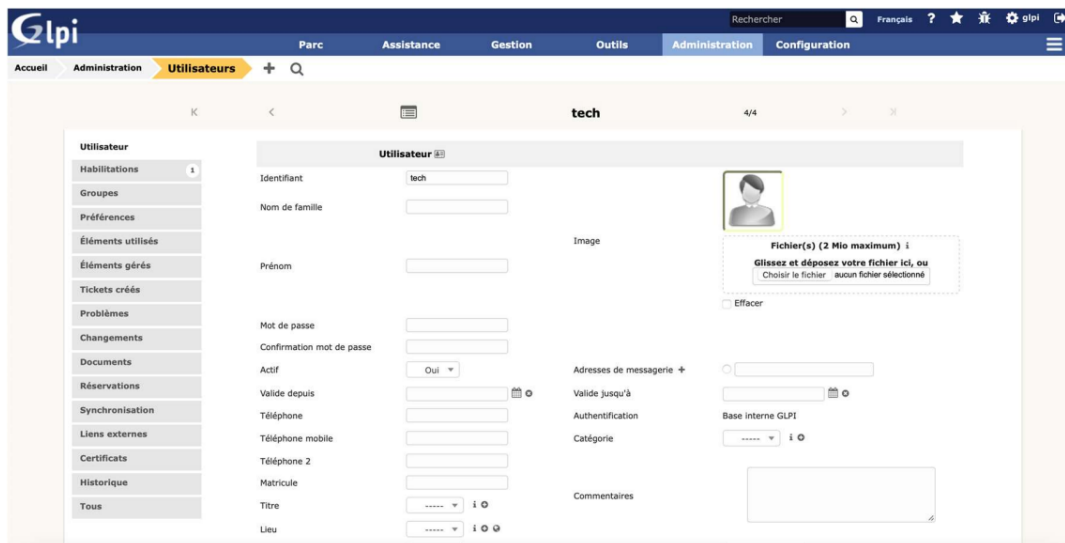
Une fois connecté, nous pouvons voir 2 messages d'alerte :



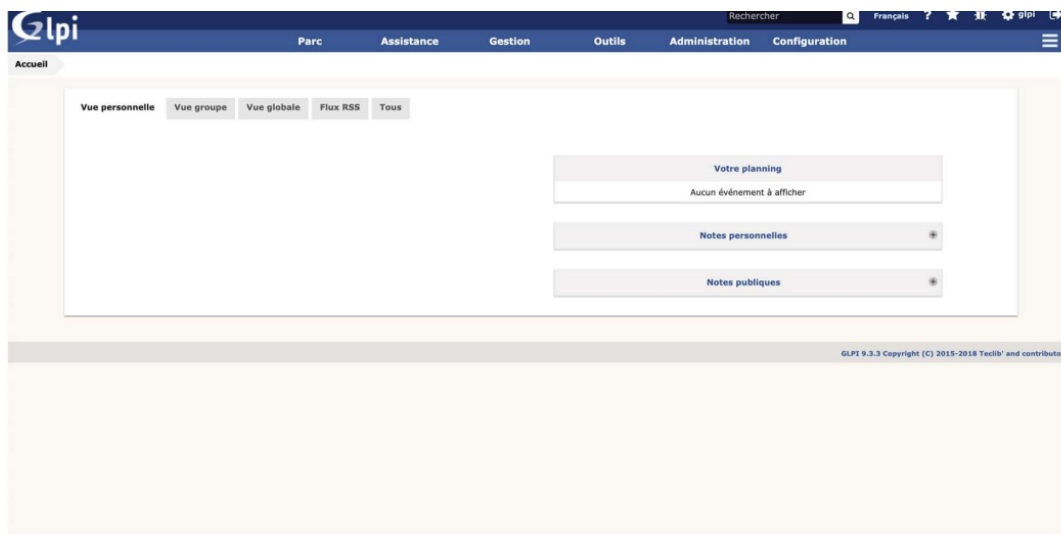
Pour des raisons de sécurité nous allons donc modifier le mot de passe par défaut et supprimer le dossier install/install.php Pour supprimer le dossier install/install.php il suffit d'entrer la commande cidessous :

```
rm /var/www/html/glp/installe/install.php
```

Nous devons ensuite via l'onglet administration modifier le mot de passe de tous les utilisateurs :

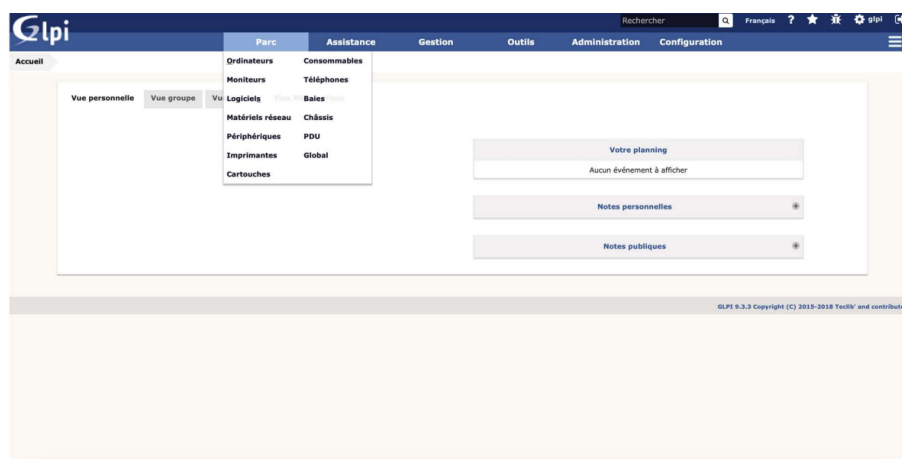


Une fois sauvegarder nous pouvons retourner sur la page d'accueil et voir que les messages d'alerte ont bien disparu, notre GLPI est donc maintenant prêt à être utilisé en toute sécurité.

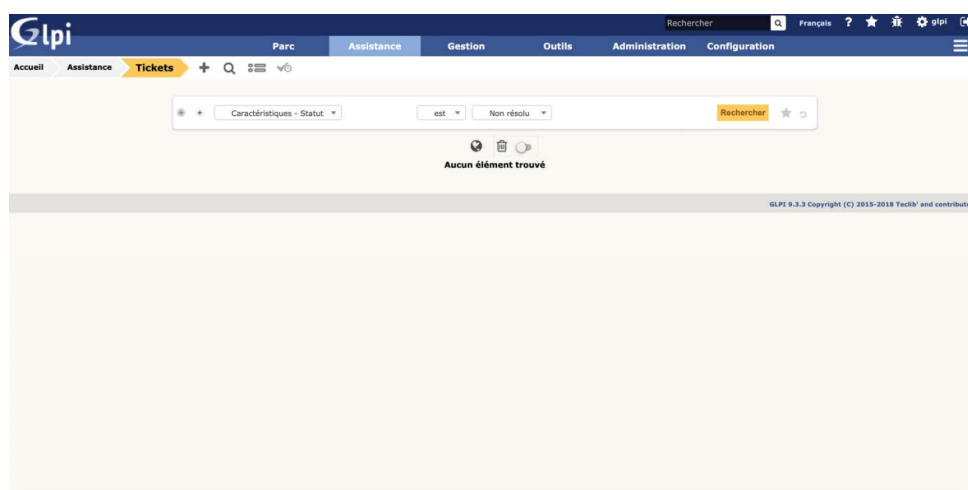


Découverte des différentes fonctions :

Nous pouvons donc maintenant ajouter tout le matériel informatique de l'entreprise via l'onglet « PARC » :



Le système de ticketing est opérationnel, nous pourrions donc maintenant ouvrir un ticket dès qu'un des salariés rencontrera un problème, il sera ensuite assigné en fonction de sa nature à l'équipe compétente.



Bilan :

Le déploiement d'un logiciel de gestion de parc ici GLPI nous a permis premièrement d'acquérir de nombreuses connaissances sur le système d'exploitation debian, l'utilisation des miroirs, les commandes sur debian, ...

Durant cette mission le choix de la solution ne nous était pas imposé nous avons donc dû effectuer des recherches sur différents logiciels proposant plus ou moins la même fonction, nous avons donc dû étudier ces logiciels pour faire un choix qui serait le plus adapté à notre entreprise et à nos besoins, cette étude des solutions fut une expérience enrichissante pour nous.

Nos recherches autour de GLPI nous ont fait comprendre l'intérêt primordial de l'implémentation d'un logiciel de gestion de parc dans une entreprise.

Nos recherches nous ont aussi permis de découvrir le logiciel GLPI en lui-même avec toutes les possibilités qu'il propose nativement comme l'ajout du matériel informatique, le système de ticketing, ...

En plus des fonctions présentes nativement, nous avons pris connaissance de certaines fonctionnalités parmi tant d'autres qui peuvent être ajoutés à GLPI via l'ajout de plugin.

La prise en main de GLPI nous a permis de gagner en compétence sur celui-ci ce qui va nous permettre par la suite son utilisation et son amélioration en fonction des besoins de l'entreprise.