16/05/2023

Enzo Nonnenmacher

CPNV

Mise en place d’un outil de ticketing

Rapport de projet

**Table des matières**

[1 Résumé du projet 3](#_Toc135137844)

[1.1 Situation de départ 3](#_Toc135137845)

[1.2 Mise en œuvre 3](#_Toc135137846)

[1.3 Résultats 3](#_Toc135137847)

[2 Analyse préliminaire 4](#_Toc135137848)

[2.1 Introduction 4](#_Toc135137849)

[*2.2* Objectifs 4](#_Toc135137850)

[*2.3* Planification initiale 5](#_Toc135137851)

[3 Analyse / Conception 6](#_Toc135137852)

[*3.1* Concept 6](#_Toc135137853)

[3.2 Stratégie de test 7](#_Toc135137854)

[*3.3* Risques techniques 8](#_Toc135137855)

[*3.4* Planification 8](#_Toc135137856)

[3.5 Dossier de conception 9](#_Toc135137857)

[4 Réalisation 9](#_Toc135137858)

[*4.1* Dossier de réalisation 9](#_Toc135137859)

[4.2 Analyse de l’outil 10](#_Toc135137860)

[4.2.1 Plugins 10](#_Toc135137861)

[4.2.2 Avantages/Inconvénients 11](#_Toc135137862)

[4.3 Configuration du serveur Debian 12](#_Toc135137863)

[4.4 Configuration du serveur Windows 13](#_Toc135137864)

[4.5 Configuration de l’active directory 15](#_Toc135137865)

[4.5.1 Unité d’organigramme 17](#_Toc135137866)

[4.5.2 Création des adresses mail utilisateurs 18](#_Toc135137867)

[4.6 Mise en place de GLPI 18](#_Toc135137868)

[4.6.1 Installation serveur LAMP 18](#_Toc135137869)

[4.6.2 Création de la base de données GLPI 19](#_Toc135137870)

[4.6.3 Installation de l’outil 19](#_Toc135137871)

[4.6.4 Lancement de GLPI 19](#_Toc135137872)

[4.7 Annuaire LDAP 22](#_Toc135137873)

[4.8 Base de connaissances 24](#_Toc135137874)

[4.9 Configuration de l’adresse mail 24](#_Toc135137875)

[4.9.1 Configuration du collecteur 24](#_Toc135137876)

[4.9.2 Configuration des notifications par mail 27](#_Toc135137877)

[4.10 Configuration du SLA 28](#_Toc135137878)

[4.10.1 Notification du dépassement du SLA 30](#_Toc135137879)

[4.11 Profils 31](#_Toc135137880)

[4.12 Rapport pour les responsables 31](#_Toc135137881)

[4.13 Automatisions des actions automatiques 33](#_Toc135137882)

[4.14 Mise en place du support 34](#_Toc135137883)

[4.14.1 Création des catégories 34](#_Toc135137884)

[4.14.2 Notifications 35](#_Toc135137885)

[4.15 Description des tests effectués 35](#_Toc135137886)

[*4.16* Erreurs restantes 35](#_Toc135137887)

[*4.17* Liste des documents fournis 35](#_Toc135137888)

[5 Conclusions 35](#_Toc135137889)

[6 Annexes 36](#_Toc135137890)

[*6.1* Résumé du rapport du TPI / version succincte de la documentation 36](#_Toc135137891)

[*6.2* Sources – Bibliographie 36](#_Toc135137892)

[*6.3* Journal de travail 36](#_Toc135137893)

[*6.4* Manuel d'Installation 36](#_Toc135137894)

[*6.5* Manuel d'Utilisation 36](#_Toc135137895)

[*6.6* Archives du projet 36](#_Toc135137896)

# Résumé du projet

## Situation de départ

## Mise en œuvre

## Résultats

# Analyse préliminaire

## Introduction

Le but de ce projet est de mettre en place l'outil GLPI en tant que solution de ticketing sur un serveur dédié. Cette solution permettra de centraliser la gestion des demandes d'intervention et des incidents, ainsi que de suivre leur résolution de manière efficace. L'interface web de GLPI permettra également de faciliter la communication entre les différents acteurs impliqués dans un ticket sur GLPI.

Dans ce rapport de projet, nous examinerons les différentes étapes nécessaires à la mise en place de GLPI, de l'installation du serveur à la configuration de l'outil. Nous aborderons également les avantages et les inconvénients de la nouvelle version GLPI et des raisons qui poussent à faire une migration du logiciel vers la dernière version.

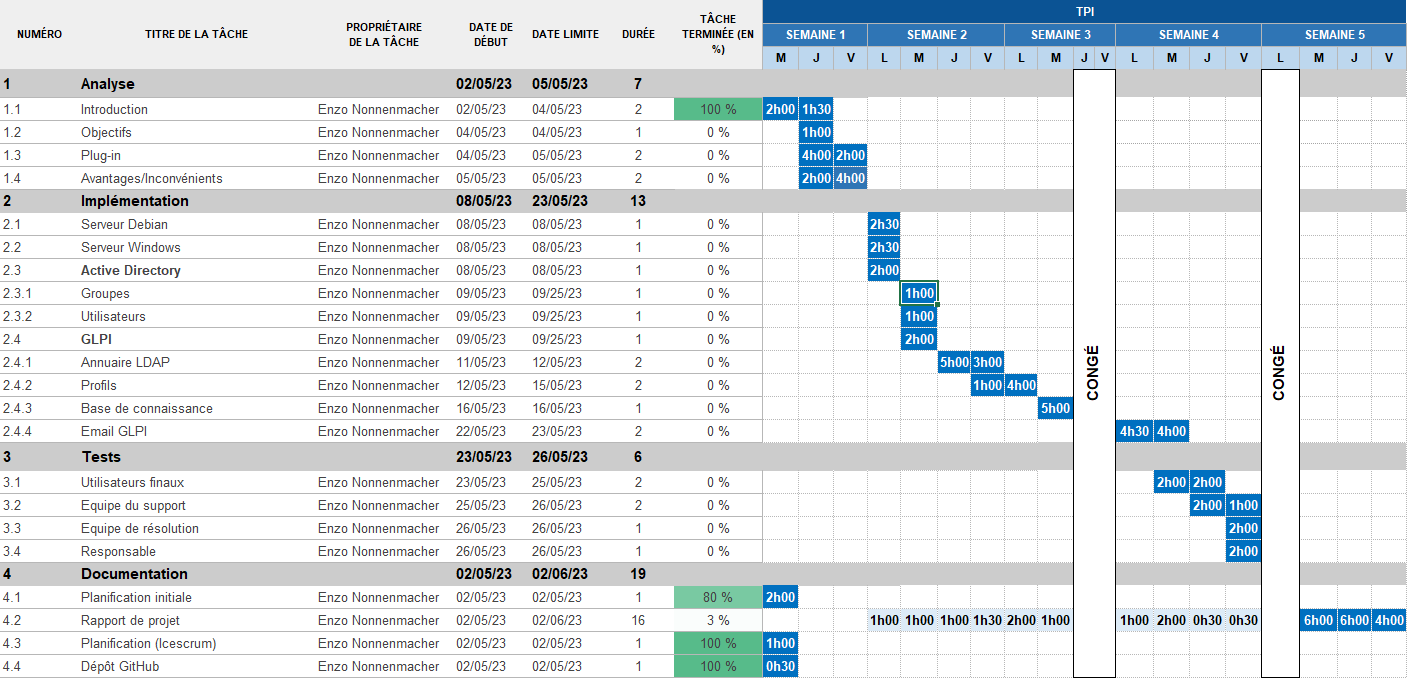
## Objectifs

L’objectif de ce projet, est d’analyser la dernière version de GLPI afin de définir les avantages et inconvénients d’utiliser cet outil, de configurer un serveur et de mettre en place GLPI dessus. Une fois cette opération faite, configurer diverse fonctionnalité de façon à ce qu’il devienne opérationnel dans le cadre d’une entreprise.

## Planification initiale

Pour la planification initiale de mon projet, j’ai décidé de faire un diagramme de Gantt. Je l’ai découpé en quatre parties importantes :

* Analyse
* Implémentation
* Tests
* Documentation

Voici le diagramme représentant ma planification initiale :

# Analyse / Conception

## Concept

Dans un premier temps, il va falloir effectuer une analyse de la dernière version de GLPI et de tout ce qui est possible de faire. Définir les différents avantages et inconvénients et visualiser les différents plugins qui pourraient être utile afin de configurer au mieux l’outil.

Une fois l’analyse de GLPI terminé, je vais configurer un serveur Debian 11 et un serveur Windows 2022. Une fois les deux serveurs prêts à l’emploi, je vais installer, dans un premier temps, un Active Directory sur le serveur Windows. Je créerai par la suite quatre groupes différents avec deux utilisateurs différents dedans. Chaque groupe aura son rôle dans GLPI.

Dès que l’active directory est configuré, j’installerai la version 10.0.6 de GLPI sur le serveur Debian en suivant une procédure trouvée sur internet.

Une fois que GLPI est utilisable, je vais implémenter différentes fonctionnalités afin de faciliter son utilisation et sa gestion. Je configurerai, en premier lieu, l’annuaire LDAP pour que je puisse intégrer les différents utilisateurs et groupes sur GLPI. J’attribuerai ensuite les droits demandés aux profils respectifs afin de les assigner au bon groupe et aux bons utilisateurs.

Je rédigerai par la suite, une dizaine de base de connaissances directement depuis GLPI. Une fois cette opération faite, je configurerai l’adresse mail sur GLPI de façon à ce qu’on puisse créer, traiter ou clôturer un ticket en passant par les mails plutôt que l’interface Web de GLPI. Je passerai ensuite à la configuration des différentes SLA.

Une fois toutes les fonctionnalités configurées et fonctionnelles, j’effectuerai une série de 13 tests avec les différents types d’utilisateurs manuellement.

## Stratégie de test

Pour ma stratégie de test, j’ai décidé d’effectuer 13 tests unitaires afin de déterminer le bon fonctionnement de GLPI suivant les indications qui m’ont été demandé dans mon cahier des charges.

Voici un tableau démontrant les tests que je vais effectuer :

|  |  |  |
| --- | --- | --- |
| Testeur (Groupes) | Action | Résultat attendu |
| Utilisateurs finaux | L’utilisateur crée un ticket depuis l’interface Web | Le ticket peut être aperçu par tout le monde sur l’interface Web |
| Utilisateurs finaux | L’utilisateur envoie un mail au support | Un ticket est créé automatiquement |
| Utilisateurs finaux | L’utilisateur envoie un mail au support | Un mail automatique lui est envoyé pour lui confirmer la création du ticket |
| Utilisateurs finaux | L’utilisateur crée un ticket depuis l’interface Web sans rentrer un champ obligatoire | Un message d’erreur apparaît qu’il manque des informations |
| Equipe du support | L’utilisateur modifie une information du ticket | Les personnes concernées par ce ticket sont informé par mail |
| Equipe du support | L’utilisateur modifie une base de connaissances | La base de connaissances est mis à jour |
| Equipe du support | L’utilisateur lie un ticket à une base de connaissances | Le ticket est lié à la base de connaissances |
| Equipe du support | L’utilisateur clique sur le bouton m’associer sur un ticket sous attribution | Le ticket est attribué à l’utilisateur |
| Equipe de résolution | L’utilisateur modifie une base de connaissances | La base de connaissances est mis à jour |
| Equipe de résolution | L’utilisateur lie un ticket à une base de connaissances | Le ticket est lié à la base de connaissance |
| Responsable | L’utilisateur clique sur le bouton tableau de bord | La page du tableau de bord apparaît |
| Responsable | L’utilisateur est le responsable d’un ticket qui a dépassé le temps du SLA | L’utilisateur est informé de la situation par mail |
| Responsable | L’utilisateur change le technicien attribué au ticket | Le ticket est attribué au nouveau technicien |

## Risques techniques

Le risque technique qui me semble être le plus problématique durant mon projet et la configuration de l’adresse mail sur GLPI. Je pense que je vais avoir beaucoup de difficulté avec cette partie car je sais qu’il y a beaucoup de configuration à faire que c’est assez complexe. Afin d’éviter au mieux ce risque, je vais me renseigner sur des forums GLPI pour avoir toutes les informations nécessaires avant de faire quoi que ce soit.

Je pense avoir également un manque de connaissances en ce qui concerne les différents SLA. Étant un domaine que je n’ai jamais étudié, je vais essayer de visualiser comment cela fonctionne exactement afin de pouvoir rendre un travail opérationnel.

## Planification

Pour la gestion de mon projet, j’ai décidé d’utiliser la méthode agile via l’outil Icescrum.

Je vais découper mon projet en 3 parties distinctes :

La première partie sera focalisée sur l’analyse du logiciel. Cette partie commencera le 2 mai 2023 et se terminera le vendredi 5 mai 2023.

La seconde partie sera focalisée sur l’implémentation des différents serveurs et des différentes fonctionnalités. Cette partie commencera le lundi 8 mai 2023 et se terminera le mardi 23 mai.

La troisième partie sera centralisée sur tous les tests que j’effectuerai sur GLPI. Cette partie commencera le jeudi 25 mai et se terminera le 2 juin.

Toute la documentation du projet se fera petit à petit tout au long du projet.

Voici le lien de mon Icescrum pour plus d’information :

<https://icescrum.cpnv.ch/p/TPIENZO/#/project>

## Dossier de conception

* Ordinateur type CPNV sous Windows 10
* VMWare Workstation pro version 17
  + Serveur GLPI
    - Système d’exploitation : Linux (Debian 11)
    - Serveur Web : Apache
    - Base de donnée : MariaDB
    - PHP 8.2
  + Serveur AD
    - Système d’exploitation : Windows server 2022
* GLPI version 10.0.6
* Office 2016
* GitHub

# Réalisation

## Dossier de réalisation

*Décrire la réalisation "physique" de votre projet*

* *les répertoires où le logiciel est installé*
* *la liste de tous les fichiers et une rapide description de leur contenu (des noms qui parlent !)*
* *les versions des systèmes d'exploitation et des outils logiciels*
* *la description exacte du matériel*
* *le numéro de version de votre produit !*
* *programmation et scripts: librairies externes, dictionnaire des données, reconstruction du logiciel - cible à partir des sources.*

## Analyse de l’outil

### Plugins

Pour effectuer une analyse des différents plug-in disponible sur GLPI, je me suis rendu sur le site GLPI Project. Ce site propose tous les plugins que la communauté à coder et qui ont mis leur création sur ce site pour que chaque utilisateur de GLPI puisse les utiliser. (Lien du site : <https://plugins.glpi-project.org/>)

Suite à cette analyse, j’ai noté 4 plugins compatible avec la version 10, qui pourrait être intéressent pour l’utilisation ou pour la configuration de GLPI.

Voici les plug-in que j’ai trouvé :

|  |  |
| --- | --- |
| Nom du plug-in | Description du plug-in |
| TicketMail | TicketMail permet d'envoyer les informations de suivi et tâches d'un ticket à une personne par mail depuis un onglet affiché dans la page descriptive d'un ticket |
| Timelineticket | Timelineticket permet d'ajouter un onglet chronologie sur les tickets. C’est-à-dire qu’il ajoute un historique et des statistiques sur les différents statuts d’un ticket, les différents techniciens et groupes qui ont été assigné sur un ticket. Il permet également d’avoir un rapport sur les temps passés sur les tickets clos |
| More reporting | More reporting contient un ensemble de nouveaux rapports statistiques. Par exemple, il permet d’avoir un rapport sur l’ancienneté d’un ticket, des rapports SLA, top catégories utilisé pour les tickets et autres. |
| Escalade | Escalade permet de simplifier l'escalade de ticket vers des groupes différents. Cela ajoute également quelques fonctionnalités comme l’affichage d’un historique visuel de l’assignation des groupes sur un ticket, le clonage rapide d’un ticket, clôture des tickets clonés en même temps et bien d’autres. |

J’avais également trouver d’autres plugins qui auraient pu être intéressant mais qui sont malheursement compatible qu’avec la version 9 de GLPI.

### Avantages/Inconvénients

Afin de pouvoir définir si passer à la dixième version de GLPI, nous allons regarder les avantages et inconvénients en comparant la version 9 et la version 10 de GLPI.

La version 10 de GLPI peut offrir une amélioration de la sécurité, des performances et des fonctionnalités, ainsi que des corrections de bugs. Un autre avantage que l’on peut noter, c’est l’aspect visuel de la version 10 de GLPI qui est plus simple d’utilisation. Le plus gros avantage et l’argument qui pousse à faire GLPI à la version 10 est le fait que la version 9 n’aura bientôt plus de support officiel de la part des développeur (Effectif le 30 juin 2023).

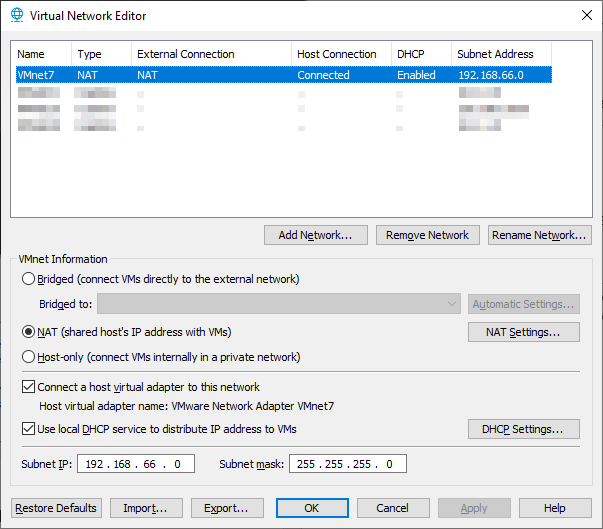
En revanche, le plus gros inconvénient de la version 10 de GLPI est son manque de disponibilité des différents plugins contrairement à la version 9 où tous les plugins sont compatibles avec cette version.

Voici un tableau représentant les avantages et les inconvénients de la version 10 de GLPI citer au-dessus :

|  |  |
| --- | --- |
| Avantages | Inconvénients |
| Amélioration de la sécurité | Manque de compatibilité avec beaucoup de plugins |
| Amélioration de la performance |  |
| Nouvelle fonctionnalité |  |
| Interface utilisateur |  |
| Correction de multiple bug |  |

## Configuration du serveur Debian

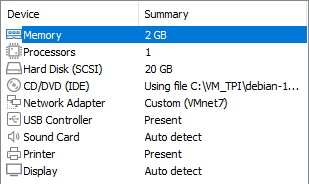
Dans un premier temps, j’ai configuré le vmnet 7 de façon à ce qu’il soit de type NAT. Je me servirai de ce vmnet pour ensuite placer mes 2 serveurs sur le même réseau.



## 

J’ai ensuite créer une nouvelle machine virtuelle avec la version 11.7.0 de Debian.

Voici les paramètres de mon serveur Debian :



Une fois la machine virtuelle crée, je l’ai démarré et j’ai lancer une installation sans l’interface graphique.

Voici les étapes que j’ai suivi pour l’installation :

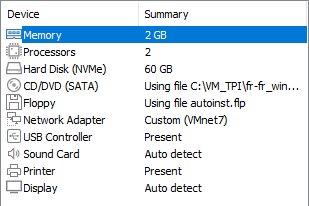
* Langue : Français
* Pays géographique : Suisse
* Région : Suisse romand
* Nom du système : Serveur-GLPI
* Domaine : Aucun
* Nom de l’utilisateur : user
* Méthode de partitionnement : Assisté – uniquement un disque entier
* Schéma de partitionnement : Tout dans une seule partition
* Pays miroir : Suisse
* Miroir de l’archive : deb.debian.org
* Mandataire http : Aucun
* Logiciel à installer :
  + Serveur SSH
  + Utilitaires usuels du système
* Installer GRUB : Oui
* Périphérique d’installation : /dev/sda

Pour la suite du projet, je ferai les différentes manipulations sur le serveur Debian via l’application cmder en connexion ssh.

## Configuration du serveur Windows

Tout d’abord, il a fallu que je créer la machine virtuelle avec le système d’exploitation Windows server 2022.

Voici les paramètres de mon serveur Windows :

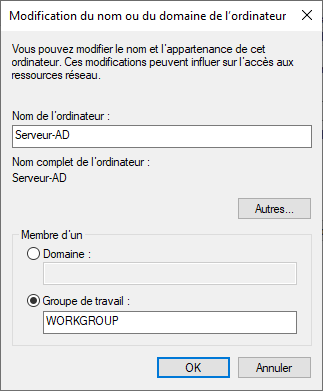


Une fois la création de la machine virtuelle terminé, je l’ai démarré afin de lancer l’installation Windows.

J’ai choisi d’installer la version Windows server 2022 standard car je ne pense pas avoir besoin de la version Datacenter sachant que ce serveur va accueillir uniquement un active directory.

Une fois que l’installation est terminé, je suis allé activer le compte administrateur local et j’ai supprimer le compte généré par VMWare.

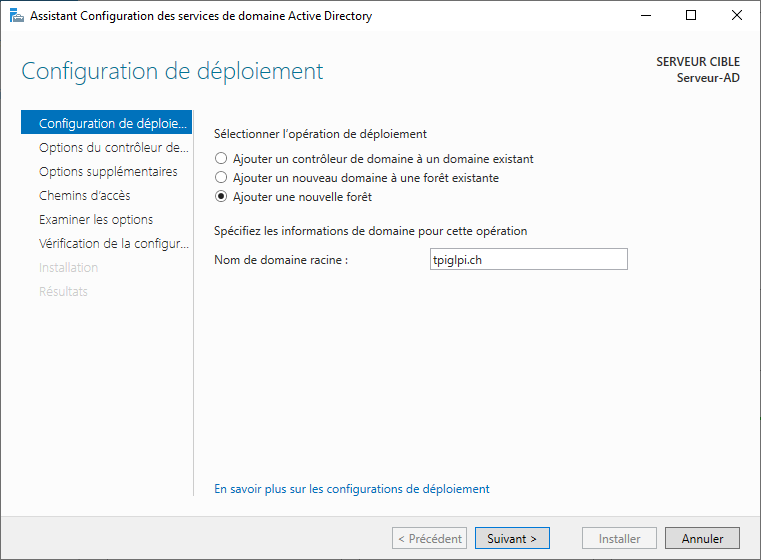
Pour finaliser la configuration du serveur, je l’ai renommé :

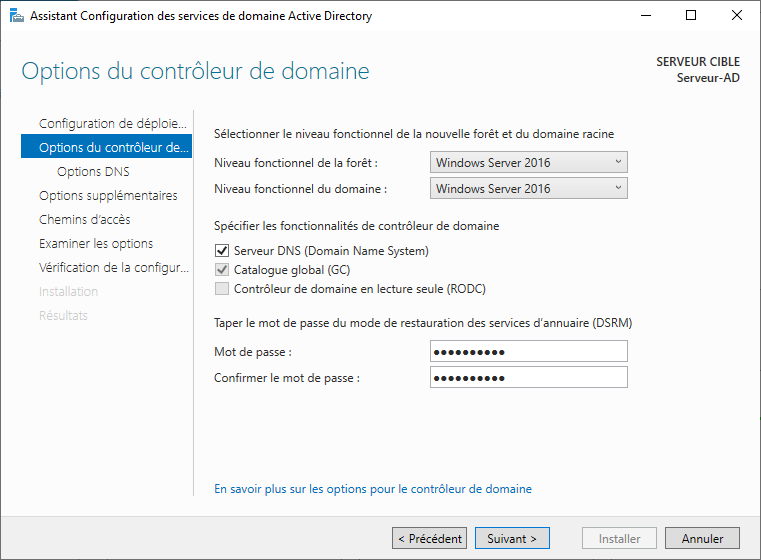


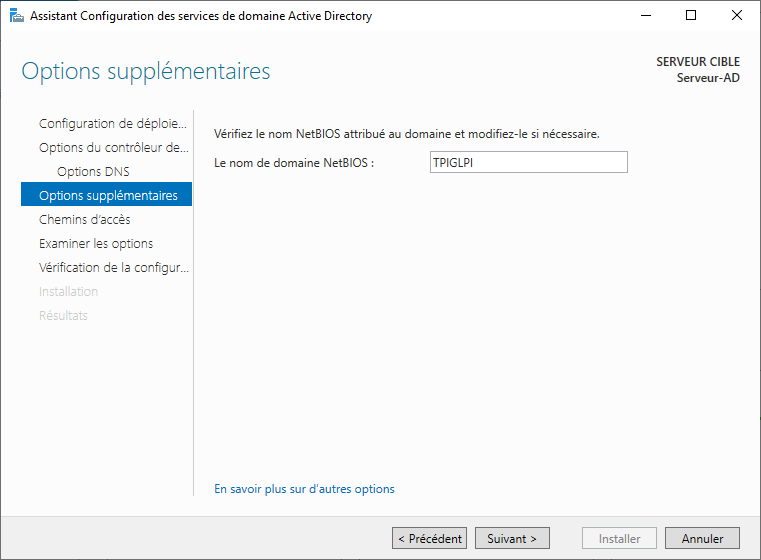
## Configuration de l’active directory

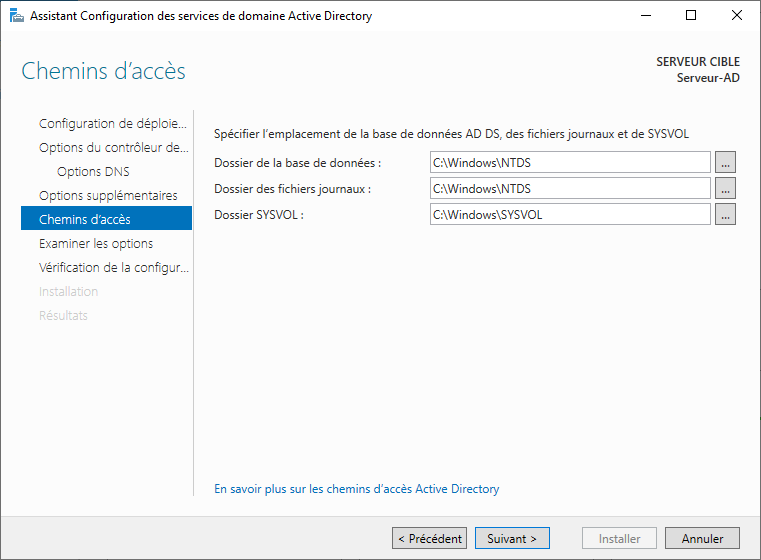
En premier lieu, j’ai ajouté le service AD DS sur mon serveur Windows

Je l’ai ensuite promu en contrôleur de domaine. Voici les étapes que j’ai suivi pour promouvoir ce serveur :







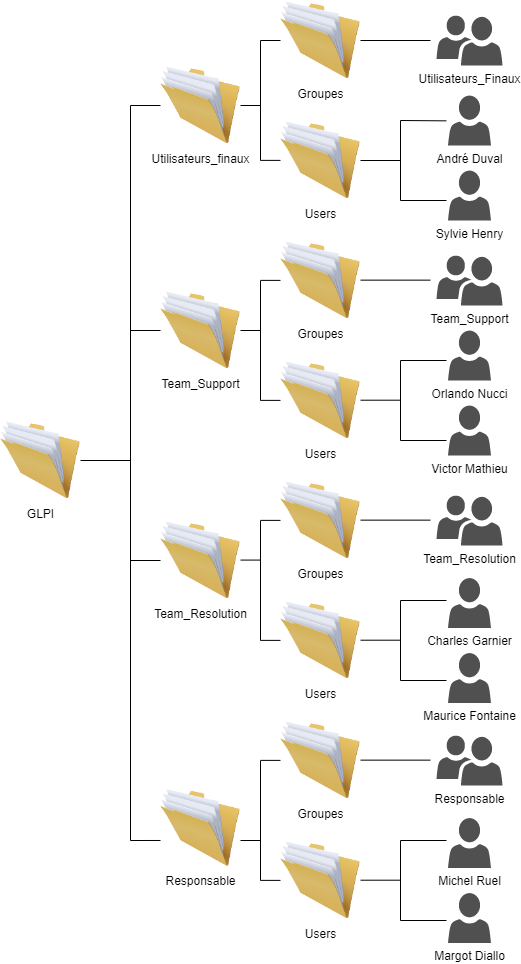


Une fois les informations remplies, j’ai installé la configuration sur mon serveur et je l’ai redémarré.

### Unité d’organigramme

Une fois le serveur redémarré, je dois créer une unité d’organigramme comportant mes différents groupes et utilisateurs pour les connecter sur GLPI par la suite.

Voici un diagramme représentant mon unité d’organigramme :



### Création des adresses mail utilisateurs

Maintenant que tous les utilisateurs sont créés et placé correctement dans l’active directory, je me suis rendu sur Gmail afin de créer les 8 adresses mails pour chaque utilisateur. J’ai décidé d’utiliser Gmail car je trouve qu’il est assez simple à utiliser et aussi pour créer des comptes Google.

Pour toutes les adresses mails, j’ai suivi la syntaxe suivante :

* [Prénom.nom.tpi@gmail.com](mailto:Prénom.nom.tpi@gmail.com)

J’ai rencontré pas mal de difficultés pour créer ces adresses mails car Google demande un numéro de téléphone pour la vérification si nous sommes bien des humains sauf qu’au bout d’un certain moment, Google bloque l’accès si on utilise plusieurs fois le même numéro.

J’ai réussi à corriger ce problème grâce à l’un de mes camarades de classe qui m’a autorisé à utiliser son numéro de téléphone.

## Mise en place de GLPI

Pour l’installation de GLPI sur mon serveur Debian, j’ai suivi une documentation que j’ai trouvé sur internet (lien du site : <https://tutos-info.fr/?s=glpi+10>)

### Installation serveur LAMP

Dans un premier temps, il faut installer le service Apache sur notre serveur avec la commande :

* apt install apache2

Une fois l’installation du service Apache terminé, j’ai installé le module PHP 8.2. La version 11 de Debian n’a pas de module PHP 8.2 par défaut d’installer. Il faut donc installer certaines dépendances grâce à cette commande :

* apt install ca-certificates apt-transport-https software-properties-common wget curl lsb-release –y
* curl -sSL https://packages.sury.org/php/README.txt | bash -x

Une fois les dépendances installées, on peut installer PHP avec cette commande :

* apt install php8.2 libapache2-mod-php8.2

Puis j’ai redémarré le service Apache pour qu’il prenne les modifications apportées avec la commande :

* systemctl restart apache2

Pour finir, il faut installer la base de données MariaDB avec ces commandes :

* apt install mariadb-server
* mysql\_secure\_installation

### Création de la base de données GLPI

Pour commencer, nous nous connectons à MariaDB grâce à la commande :

* mysql –u root –p

Une fois connecté, voici les commandes que j’ai utilisé pour créer une base de données fonctionnel avec un compte ayan tous les droits dessus :

* create database glpi;
* create user 'glpi'@'localhost' identified by 'glpi';
* grant all privileges on glpi.\* to 'glpi'@'localhost' with grant option;
* flush privileges;
* quit

### Installation de l’outil

Maintenant que l’environnement est prêt à accueillir GLPI, nous allons exécuter la commande suivant afin d’installé le zip de la version 10.0.6 de GLPI sur notre serveur :

* wget https://github.com/glpi-project/glpi/releases/download/10.0.6/glpi-10.0.6.tgz

Une fois téléchargé, il faut déziper ce dossier avec la commande :

* tar xvf glpi-10.0.6.tgz

Après cela, il faut déplacer le dossier glpi est le mettre dans l’arborescence d’Apache /var/www/html grâce à cette commande :

* mv glpi /var/www/html/glpi

Pour finir, on peut supprimer l’ancien dossier zip avec cette commande :

* rm glpi-10.0.6.tgz

### Lancement de GLPI

Avant de lancer GLPI, il a quelques autres modules qui doivent être installé afin d’assurer le bon fonctionnement de GLPI. Voici la commande pour effectuer ces installations :

* apt install php8.2-curl php8.2-gd php8.2-mbstring php8.2-zip php8.2-xml php8.2-ldap php8.2-intl php8.2-mysql php8.2-dom php8.2-simplexml php-json php8.2-phpdbg php8.2-cgi

Autre point important, il faut apporter des modifications sur propriété du dossier qu’on donne à l’administrateur d’Apache avec ces commandes :

* chown -R www-data:www-data /var/www/html/glpi/
* chmod -R 755 /var/www/html/glpi/

Puis on redémarre le service apache :

* systemctl restart apache2

Maintenant, on peut accéder à l’interface Web de GLPI en tapant l’adresse IP du serveur /glpi dans un navigateur.

Maintenant que nous somme sur l’interface Web, on peut sélectionner la langue française, puis cliquer sur installer GLPI.

Une fois qu’on arrive sur la page où cela demande de renseigner les informations de connexion à la base de données, voici comment j’ai renseigné les informations :



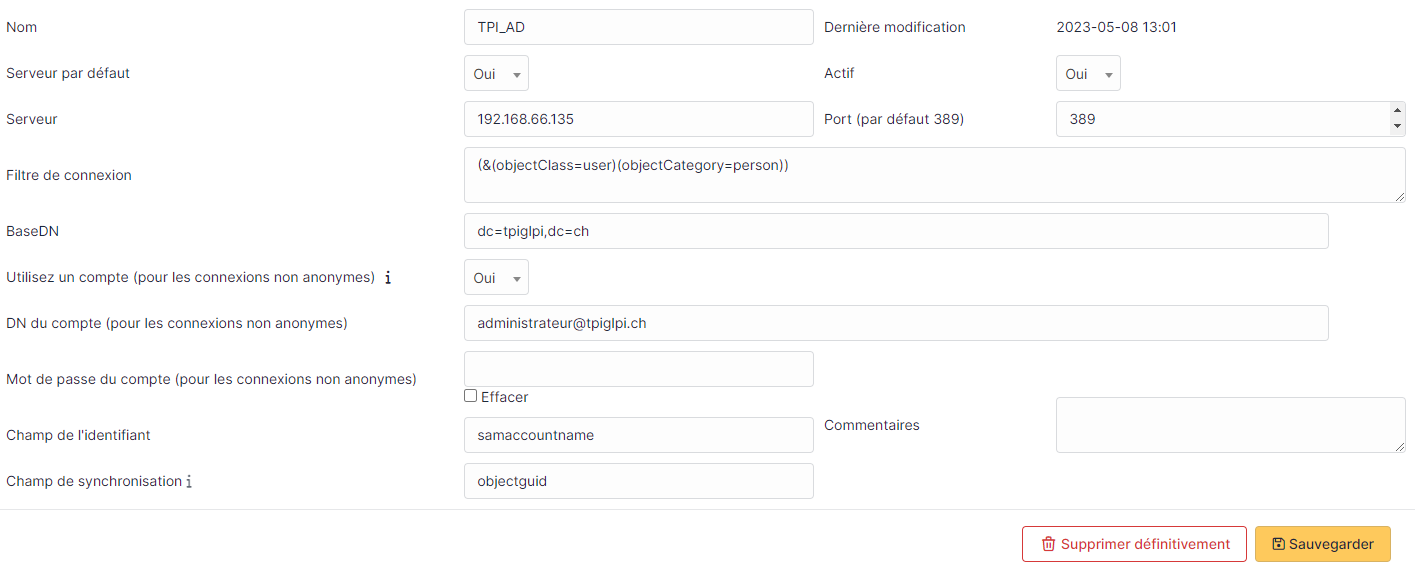
Il indique ensuite que la connexion à la base de données a été réussi et on peut choisir la base de données glpi puis continuer :



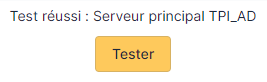
## Annuaire LDAP

Pour ajouter un annuaire LDAP sur GLPI, il faut vous rendre sur Configuration > Authentification puis cliquer sur Annuaire LDAP.

Voici les informations que j’ai saisi pour configurer mon annuaire :



Pour tester que la connexion au serveur AD fonctionne, on clique sur l’onglet tester sur la configuration de l’annuaire, puis on clique sur le bouton tester et il affiche ce message :

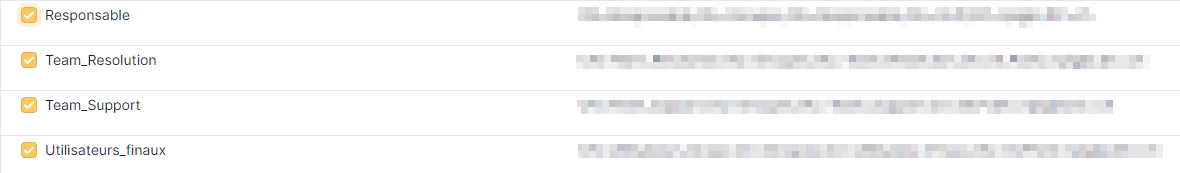


Une fois que la configuration de l’annuaire est fini, on peut importer les différents groupes et utilisateurs provenant de l’active directory :

Pour importer les utilisateurs, on se dirige dans l’onglet Administration > Utilisateurs, on clique ensuite sur le bouton Liaison annuaire LDAP > Importation de nouveau utilisateurs. On recherche sans taper d’information et on coche les utilisateurs que l’on veut importer :



Pour l’importation des groupes, c’est exactement la même opération que pour les utilisateurs mais faut se rendre sur l’onglet Administration > Groupes :



Attention, si les utilisateurs sont importés avant les groupes, il faut effectuer une synchronisation de tous les comptes qui ont été ajouté précédemment.

## Base de connaissances

Suite à la demande de mon cahier des charges, j’ai créé une dizaine de bases de connaissances en y ajoutant une catégorie à chaque fois afin de pouvoir différencier les types de base de connaissances.

Voici un tableau représentant mes bases de connaissances :

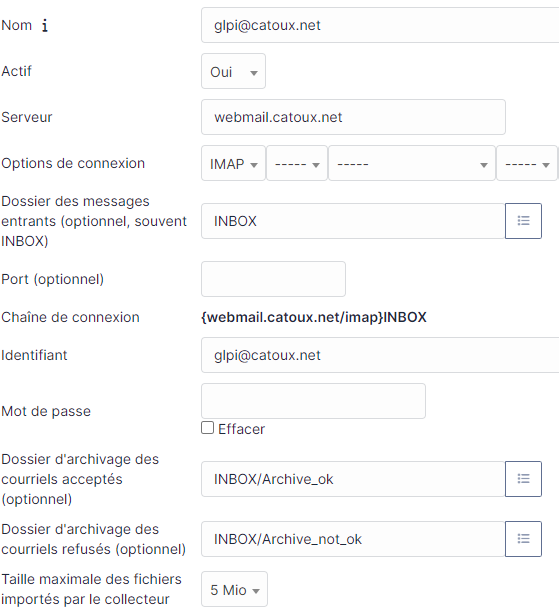
|  |  |
| --- | --- |
| Titre | Catégorie |
| Création d’une base de connaissance | Aide GLPI > Outils |
| Ajouter un suivi sur un ticket depuis l’interface Web | Aide GLPI > Ticket |
| Assigner un gabarit à une catégorie de ticket | Aide GLPI > Ticket |
| Clôturer un ticket depuis l’interface Web | Aide GLPI > Ticket |
| Mettre à la corbeille un ticket | Aide GLPI > Ticket |
| Supprimer définitivement un ticket | Aide GLPI > Ticket |
| Création d’un ticket depuis l’interface Web | Aide GLPI > Ticket |
| Synchroniser un ensemble d’utilisateurs | Aide GLPI > User/Group |
| Importer des groupes sur GLPI | Aide GLPI > User/Group |
| Importer des utilisateurs sur GLPI | Aide GLPI > User/Group |

## Configuration de l’adresse mail

### Configuration du collecteur

Je me suis rendu, sur l’onglet Configuration > Collecteur afin de configurer le collecteur par mail. Les informations du serveur et de la boîte mail m’ont été prescris par ma chef de projet.

Voici la configuration de mon collecteur :



Une fois que le collecteur est configuré, il faut tester la connexion, pour cela, je me suis rendu sur l’onglet Action afin de pouvoir synchroniser mon collecteur de mail.

Pour voir si mon collecteur fonctionnait ou non, j’ai essayé d’envoyer un mail via mon adresse mail du CPNV et forcer la récupération des mails depuis les paramètres du collecteur. Le mail n’est même pas apparu dans les mails non-importés par GLPI

Le test n’ayant pas fonctionné, j’ai analyser que le fuseau horaire de mon serveur n’était pas correct, j’ai changé l’heure depuis mon serveur Debian et j’ai essayé de refaire le test. Il indique qu’il refuse l’importation du mail.

Le collecteur ne fonctionnant toujours pas, je suis allé sur différents forums afin de trouver une possible solution. J’ai trouvé qu’il fallait allait changés les paramètres de création de ticket dans les paramètres d’assistance sur GLPI

J’ai donc essayé cette solution est le collecteur fonctionne.

Maintenant que le collecteur est opérationnel, il faut que j’automatise mon collecteur de façon à ce que toutes les minutes, il récupère les mails se trouvant sur la boîte mail.

Pour ce faire, il suffit de se rendre dans l’onglet Configuration > Actions automatiques et modifier l’action nommé mailgate :



Une fois les modifications terminées, il faut se rendre sur le serveur Debian et configuré l’action automatique également sur le serveur. Voici les commandes qui m’ont permis de faire ça :

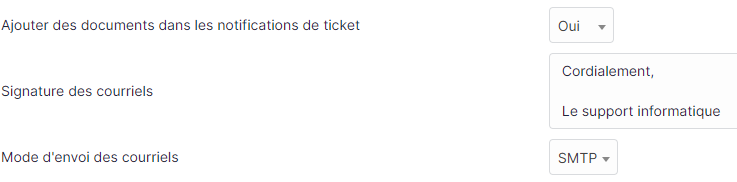
* crontab -e
* \* \* \* \* \* php /var/www/html/glpi/front/cron.php --force mailgate >> /var/www/html/glpi/log\_glpi.log 2>&1

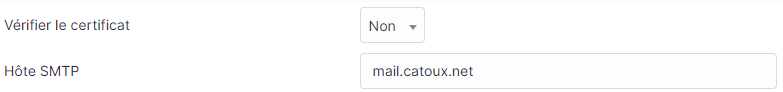
Attention, il se peut qu’après cette modification, le fuseau horaire de la tache automatique soit incorrect, pour corriger cela, il faut modifier la time zone dans le fichier php.ini (/etc/php/8.2/cli/php.ini) et mettre le fuseau horaire souhaité.

### Configuration des notifications par mail

Maintenant que notre collecteur est fonctionnel, je me suis attaqué à la seconde partie qui est la configuration des notifications par mail de la création d’un ticket, d’un suivi ou autres.

Je me suis rendu dans l’onglet Configuration > Notifications, j’ai activer le suivi par email et je suis allé configurer les notifications par mail comme ceci :









J’ai ensuite effectuer un test d’envoi sur l’adresse mail en question et cela a fonctionné

Maintenant que la configuration des notifications par courriel est terminée, je me suis rendu dans l’onglet Configuration > Notifications > Notifications et j’ai créé une nouvelle notification nommé New Ticket. Cette notification va me permettre de notifier les demandeurs du ticket par mail que le ticket a bien été créer.

Une fois la notification créer, je me suis rendu sur les gabarits de notifications et j’ai créé un gabarit que j’ai nommé de la même façon que la notification afin de m’y retrouver. Le gabarit va me servir à rédiger la forme de mon mail.

## Configuration du SLA

Avant de commencer à configurer directement le SLA, j’ai effectué une analyse du temps de prise en charge et de résolution pout chaque priorité d’un ticket.

Voici un tableau représentant mon idée du temps de prise en charge et de résolution pour chaque priorité :

|  |  |  |
| --- | --- | --- |
|  | TTO | TTR |
| Très haute | 20 min | 4h |
| Haute | 30 min | 8h |
| Moyenne | 40 min | 12h |
| Basse | 50 min | 24h |
| Très basse | 1h | 48h |

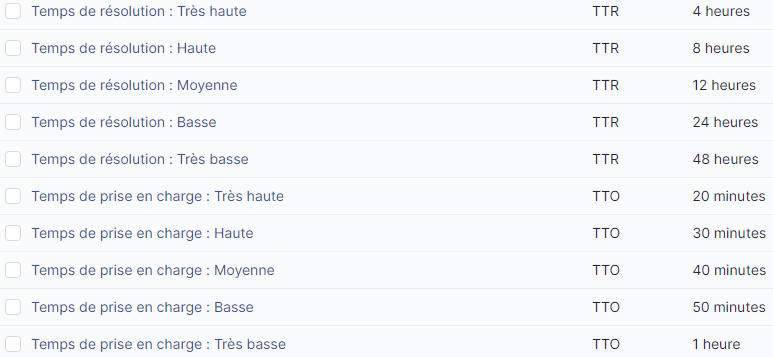
\*TTO : temps de prise en charge du ticket

\*TTR : temps de résolution d’un ticket

J’ai choisi de prendre ces valeurs car je pense que c’est une façon logique de déterminé un temps de prise en charge pratiquement équivalent pour chaque priorité mais de s’occuper au plus vite des tickets qui ont une grosse priorité et de regarder ceux qui sont moins importants plus tard.

Après l’analyse, je me suis rendu dans l’onglet Configuration > Niveau de Service et j’ai ajouté un niveau de service que j’ai appelé SLA TPI.

J’ai ensuite créer les 10 SLA avec les valeurs définis dans le tableau au-dessus comme ceci :



Une fois la création terminée, j’ai été créer les différentes règles dans l’onglet Administration > Règle > Règle métier pour les tickets, qui permettront d’attribuer les bonnes valeurs dans un ticket.

J’ai commencé par créer les 5 différents règles comme ceci :

* Priorité très haute :
  + Nom de la règle = TTR & TTO Très haute
  + Critères = Priorité est très haute
  + Action
    - SLA TTR → Temps de résolution : Très haute
    - SLA TTO → Temps de prise en charges : Très haute

* Priorité haute :
  + Nom de la règle = TTR & TTO Haute
  + Critères = Priorité est haute
  + Action
    - SLA TTR → Temps de résolution : Haute
    - SLA TTO → Temps de prise en charges : Haute
* Priorité moyenne :
  + Nom de la règle = TTR & TTO moyenne
  + Critères = Priorité est moyenne
  + Action
    - SLA TTR → Temps de résolution : Moyenne
    - SLA TTO → Temps de prise en charges : Moyenne
* Priorité basse :
  + Nom de la règle = TTR & TTO Basse
  + Critères = Priorité est basse
  + Action
    - SLA TTR → Temps de résolution : Basse
    - SLA TTO → Temps de prise en charges : Basse
* Priorité très basse :
  + Nom de la règle = TTR & TTO Très basse
  + Critères = Priorité est très basse
  + Action
    - SLA TTR → Temps de résolution : Très basse
    - SLA TTO → Temps de prise en charges : Très basse

### Notification du dépassement du SLA

Avant de configurer les notifications, il est important d’ajouter un niveau d’escalade à tous les SLA. Voici comment j’ai procédé :

* Nom : Dépassement du SLA
* Exécution : dès que le temps de résolution est atteint
* Action : Envoi un rappel automatique des SLA

C’est la même procédure pour tous les SLA de type TTR (temps de résolution).

Une fois le niveau d’escalade configuré, je me suis rendu sur l’onglet Configuration > Notification > Notification. J’ai ensuite créer une nouvelle notification que j’ai nommé SLA alerte, j’ai ajouté le groupe Responsables comme destinataire et j’ai ajouté un gabarit que j’avais créé préalablement.

## Profils

Pour la création de mes différents profils, j’ai décidé d’en créer 4 ayant les mêmes noms des différents groupes créer préalablement dans l’active directory :

* Utilisateurs finaux
* Team support
* Team résolution
* Responsables

Pour le profil des utilisateurs finaux, j’ai décidé de m’inspirer des droits d’un profil qui était déjà existant sur GLPI qui se nomme Self-Service. Basiquement, les utilisateurs finaux ont le droit de créer, ajouter un suivi et de visualiser les tickets existant. Ils n’ont pas les droits de suppression ou de modification des tickets déjà existant.

Pour les profils de l’équipe support et résolution, je me suis inspiré également d’un profil déjà existant qui se nomme Technician. J’ai choisi de mettre les mêmes droits pour ces 2 profils car ils auront les même droits et accès sur GLPI. Les membres de l’équipe support et résolution ont les mêmes droits sur les tickets mais ils peuvent les modifier à leur guise. Ils ont également des droits de création et de modification sur les bases de connaissances.

Pour finir, le profil Admin qui est déjà présent sur GLPI m’a aidé à attribuer les droits pour le profil Responsables. Les responsables auront tous les droits sur les tickets ainsi que la base de connaissances.

## Rapport pour les responsables

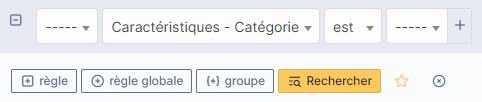
Suite à la demande sur mon cahier des charges, les 2 responsables doivent avoir 4 rapports différents disponible :

* Le nombre de tickets ouverts/fermés/en cours par catégorie
* Le nombre de tickets par technicien (Team support)
* Le nombre de tickets avec le temps de résolution dépassé
* Les logiciels ou le matériel qui a le plus de tickets ouverts

Pour chaque rapport, j’ai décidé de faire une recherche des tickets directement sur GLPI est de la mettre en favoris sur les 2 utilisateurs.

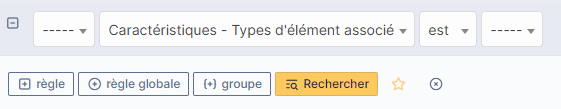
J’ai pris cette décision car je trouve que c’est une solution plus simple à utiliser qui nécessite beaucoup moins de ressource que d’exporter des rapports sachant qu’il faudrait en plus installer un plug-in sur GLPI pour effectuer cette tâche.

Voici comment j’ai procédé :



Permet de trouver le nombre de ticket ouverts/fermés/en cours par catégorie



Permet de trouver le nombre de ticket assigné au technicien souhaité

Permet de trouver le nombre de ticket qui ont dépassé le temps de résolution selon l’urgence choisi

Permet de trouver le nombre de ticket avec le type de matériel associé choisi

Pour utiliser facilement ces recherches, lorsqu’un utilisateur ayant un profil responsable, il clique sur le bouton Liste se trouvant en haut de la page et il sélectionne ensuite le rapport qu’il désire visualiser. Il suffit ensuite de sélectionner quel type de données il veut afficher.

## Automatisions des actions automatiques

Pour mon projet, j’ai besoin de rendre automatique 3 actions :

* Mailgate (collecteur de mail)
* Queuednotification (Envoi des mails au utilisateurs)
* Slaticket (Permet de notifier si le temps de résolution est dépassé)

Pour effectuer cette tâche, je suis allé sur chacune des actions et j’ai activé le mode d’exécution CLI et qu’elles s’actionnent toutes les minutes.

Une fois cette opération faite, je me suis rendu sur mon serveur debian et j’ai exécuté plusieurs commandes :

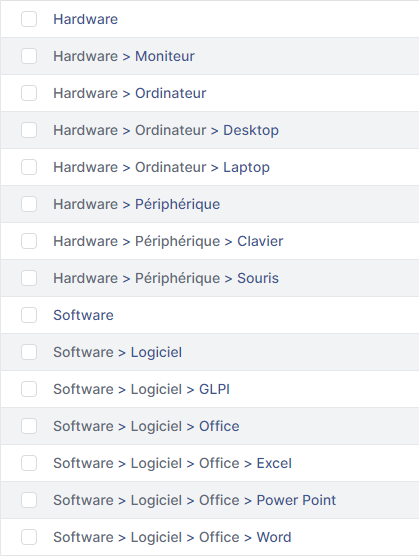
* Crontab –e
* \* \* \* \* \* php /var/www/html/glpi/front/cron.php - -force mailgate queuednotification slaticket

J’ai redémarré ensuite le service apache2 et je suis allé regarder la dernière exécution de ces actions pour vérifier que tout fonctionne correctement.

## Mise en place du support

### Création des catégories

Pour les catégories qui vont être disponible sur GLPI, j’ai décidé de créer une quinzaine de catégories. Je me suis rendu sur l’onglet Configuration > Intitulé > Assistance > Catégorie ITIL et j’ai créé mes catégories comme ceci :



J’ai décidé de créer des catégories avec une arborescence car je trouve, personnellement, que c’est beaucoup plus compréhensif avec une certaine architecture plutôt que juste du texte.

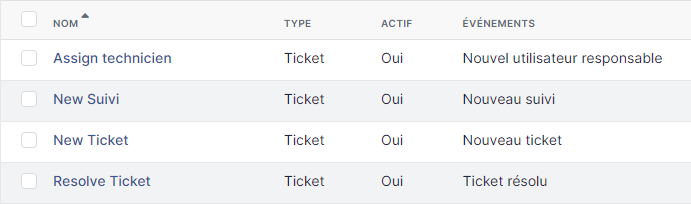
### Notifications

Afin de garantir un support optimal, je pense qu’il est nécessaire de configurer 4 notifications :

* Une notification à l’utilisateur lorsque l’utilisateur créer un ticket
* Une notification à l’utilisateur lorsque son ticket est assigné à un technicien
* Une notification à l’utilisateur lorsqu’il y a un suivi
* Une notification à l’utilisateur lorsqu’il est résolu

Suite à mon expérience dans le domaine du support, je pense qu’avertir l’utilisateur à ces moments précis est optimal afin d’avoir un contact entre le support et l’utilisateur optimal.

J’ai donc, dans un premier temps, créer mes 4 notifications comme ceci :



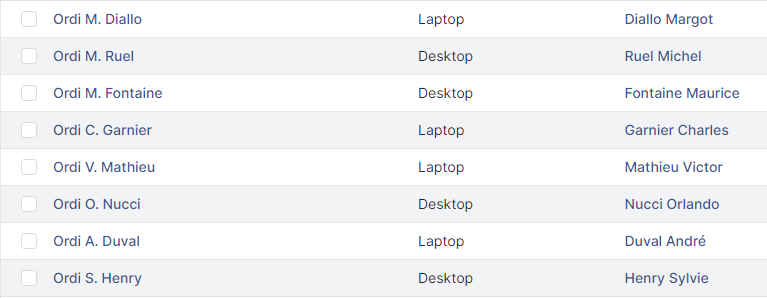
Une fois terminé, je suis allé créer les différents gabarits de chaque notifications en les nommant de la même façon que les notifications afin de ne pas faire d’erreurs.

### Création des éléments

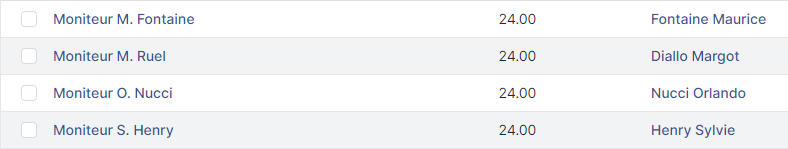
Afin de pouvoir tester le rapport des tickets lié à des éléments et aussi rendre l’utilisation de GLPI réaliste, je vais créer 8 ordinateurs et 4 moniteurs pour que je puisse les liés sur les différents tickets

J’ai décidé de faire comme cela, car j’ai créé 8 utilisateurs, il y aura donc 1 ordinateur et 1 moniteur pour les utilisateurs qui ont un desktop.

#### Ordinateurs



#### Moniteurs



## Description des tests effectués

*Pour chaque partie testée de votre projet, il faut décrire:*

* *les conditions exactes de chaque test*
* *les preuves de test (papier ou fichier)*
* *tests sans preuve: fournir au moins une description*

## Erreurs restantes

*S'il reste encore des erreurs:*

* *Description détaillée*
* *Conséquences sur l'utilisation du produit*
* *Actions envisagées ou possibles*

## Liste des documents fournis

*Lister les documents fournis au client avec votre produit, en indiquant les numéros de versions*

* *le rapport de projet*
* *le manuel d'Installation (en annexe)*
* *le manuel d'Utilisation avec des exemples graphiques (en annexe)*
* *autres…*

# Conclusions

*Développez en tous cas les points suivants:*

* *Objectifs atteints / non-atteints*
* *Points positifs / négatifs*
* *Difficultés particulières*
* *Suites possibles pour le projet (évolutions & améliorations)*

# Annexes

## Résumé du rapport du TPI / version succincte de la documentation

## Sources – Bibliographie

*Liste des livres utilisés (Titre, auteur, date), des sites Internet (URL) consultés, des articles (Revue, date, titre, auteur)… Et de toutes les aides externes (noms)*

## Journal de travail

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Durée** | **Activité** | **Remarques** |
|  |  |  |  |
|  |  |  |  |

## 

## Manuel d'Installation

## Manuel d'Utilisation

## Archives du projet

*Media, … dans une fourre en plastique*