



Gestion de tickets avec GLPI

Procédure pour Windows 10 et Debian 9

Tunui Franken



Table des matières

1	Installation	3
1.1	Avant propos	3
1.2	Matériel/logiciel à disposition	3
1.3	Installation de XAMPP sur Windows	3
1.4	Installation de XAMPP sur Debian 9	3
1.5	Activation de l’extension ‘fileinfo’	6
1.6	Installation de GLPI	7
1.7	Installation de Fusioninventory Server	8
1.8	Installation de Fusioninventory Agent	9
1.9	Affichage de l’inventaire dans GLPI	10
2	Gestion du parc	12
2.1	Changement de l’intervalle de remontée	12
2.2	Ajout manuel d’une imprimante	12
2.3	Administration de contrat	12
2.4	Remontée d’informations par SNMP	12
3	Ticketing	15
3.1	Créer des entités	15
3.2	Créer des groupes	15
3.3	Créer des comptes	16
3.4	Créer des catégories de ticket	17
3.5	Créer des tickets	17
3.6	Répondre à un ticket ou le résoudre	17
4	Sources	19

1 Installation

1.1 Avant propos

Nous proposons ici deux méthodes d'installation : sous Windows et sous Debian 9. Si rien n'est indiqué, les étapes s'appliquent aux deux.

1.2 Matériel/logiciel à disposition

- une adresse IP statique dans le réseau local du labo
- *pour Debian 9* : un utilisateur `tssr5` qui est dans le groupe `sudo`
- *pour Windows* :
 - un fichier `xampp-win32-5.6.8-0-VC11-installer.exe`
 - un fichier `glpi-0.90.tar.gz`
 - un fichier `fusioninventory-for-glpi_0.90.1.4.tar.gz`
 - un fichier `fusioninventory-agent_windows-x64_2.3.18.exe`

1.3 Installation de XAMPP sur Windows

Si vous installez sur Debian 9, ignorez cette section et rendez-vous à la prochaine section, 1.4.

Sur Windows on va utiliser le fichier `xampp-win32-5.6.8-0-VC11-installer.exe` pour installer XAMPP. En lançant l'installation on obtient une liste de composants à installer. Par défaut tout est coché.

Les composants qui nous intéressent sont :

- Apache
- MySQL
- PHP
- phpMyAdmin (optionnel mais pratique)

On peut laisser tout coché, et appuyer sur `next` jusqu'à la fin de l'installation.

1.4 Installation de XAMPP sur Debian 9

Sur Debian 9, XAMPP ne s'installe pas en paquet autonome. Nous allons donc installer séparément ses composantes.

On commence par mettre à jour la liste des paquets :

```
sudo apt update && sudo apt upgrade -y
```

1.4.1 Apache2

On installe le paquet correspondant à Apache2, et sa documentation.

```
sudo apt install apache2 apache2-doc
```

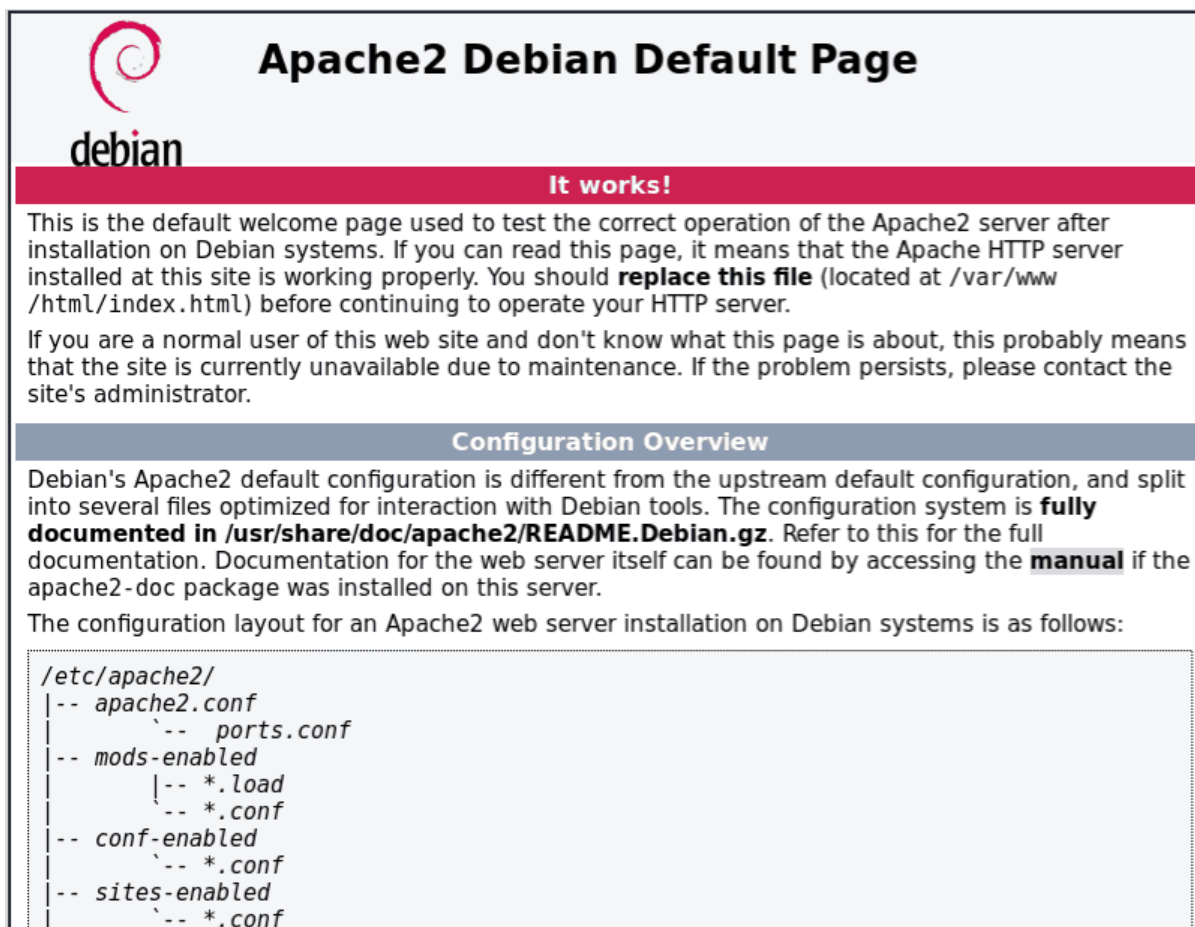
Puis on peut vérifier que le service Apache tourne :

```
sudo service apache2 status
```

S'il ne tourne pas, le lancer avec :

```
sudo service apache2 start
```

Pour être sûr que le serveur Apache2 marche, on va afficher la page d'accueil par défaut dans un navigateur avec l'adresse `http://localhost/`. On devrait obtenir la page suivante :



Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in [/usr/share/doc/apache2/README.Debian.gz](#)**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

1.4.2 MariaDB

MariaDB est un fork de MySQL créé par le créateur de MySQL lors du rachat par Oracle. Il est maintenu par une communauté et complètement compatible avec le MySQL d'Oracle. Ainsi, même si le paquet installé sera MariaDB, nous allons utiliser les commandes MySQL.

On commence par installer le paquet :

```
sudo apt install mariadb-server
```

Il faut ensuite lancer un script de post-installation important pour la sécurité sur serveur.

```
sudo mysql_secure_installation
```

Cela consiste essentiellement à ajouter un mot de passe root et à désactiver les bases de données test ainsi que l'accès à distance. On peut donc répondre par oui à toutes les questions. Quand l'utilitaire nous demande un nouveau mot de passe pour root, on peut utiliser par exemple tssr5.

Par défaut, root ne peut se connecter à MySQL que par l'utilisation de sudo. Cela va empêcher l'accès root à GLPI. On va donc créer un utilisateur glpi pour MySQL.

```
sudo mysql -u root -p
MariaDB [(none)]> CREATE USER 'glpi'@'localhost' IDENTIFIED BY 'glpi';
MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.* TO 'glpi'@'localhost';
MariaDB [(none)]> exit
```

On peut maintenant tester l'accès avec l'utilisateur que l'on vient de créer, sans l'utilisation de sudo :

```
mysql -u glpi -p
```

1.4.3 PHP

Pour PHP il faut bien-sûr installer PHP, mais aussi les modules nécessaires :

```
sudo apt install php-common libapache2-mod-php php-mysql php-mbstring php-gd
php-xml
```

On teste l'installation en créant un fichier index.php dans le répertoire /var/www/html. On va y mettre une simple fonction PHP qui nous donne des informations sur le serveur.

```
cd /var/www/html/
sudo echo '<?php<_phpinfo();<_?>' | sudo tee index.php
```

On peut regarder ce que nous renvoie cette fonction en tapant dans notre navigateur l'adresse <http://localhost/index.php>. On obtient une page comme la suivante :

PHP Version 7.0.33-0+deb9u10


System	Linux debian9vm 4.9.0-13-amd64 #1 SMP Debian 4.9.228-1 (2020-07-05) x86_64
Build Date	Oct 6 2020 17:08:28
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.0/apache2
Loaded Configuration File	/etc/php/7.0/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.0/apache2/conf.d
Additional .ini files parsed	/etc/php/7.0/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.0/apache2/conf.d/10-opcache.ini, /etc/php/7.0/apache2/conf.d/10-pdo.ini, /etc/php/7.0/apache2/conf.d/20-calendar.ini, /etc/php/7.0/apache2/conf.d/20-ctype.ini, /etc/php/7.0/apache2/conf.d/20-exif.ini, /etc/php/7.0/apache2/conf.d/20-fileinfo.ini, /etc/php/7.0/apache2/conf.d/20-ftp.ini, /etc/php/7.0/apache2/conf.d/20-gd.ini, /etc/php/7.0/apache2/conf.d/20-gettext.ini, /etc/php/7.0/apache2/conf.d/20-iconv.ini, /etc/php/7.0/apache2/conf.d/20-json.ini, /etc/php/7.0/apache2/conf.d/20-mbstring.ini, /etc/php/7.0/apache2/conf.d/20-mysqli.ini, /etc/php/7.0/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.0/apache2/conf.d/20-phar.ini, /etc/php/7.0/apache2/conf.d/20-posix.ini, /etc/php/7.0/apache2/conf.d/20-readline.ini, /etc/php/7.0/apache2/conf.d/20-shmop.ini, /etc/php/7.0/apache2/conf.d/20-sockets.ini, /etc/php/7.0/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.0/apache2/conf.d/20-sysvsem.ini, /etc/php/7.0/apache2/conf.d/20-sysvshm.ini, /etc/php/7.0/apache2/conf.d/20-tokenizer.ini
PHP API	20151012
PHP Extension	20151012
Zend Extension	320151012
Zend Extension Build	API320151012,NTS
PHP Extension Build	API20151012,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv2, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:
 Zend Engine v3.0.0, Copyright (c) 1998-2017 Zend Technologies
 with Zend OPcache v7.0.33-0+deb9u10, Copyright (c) 1999-2017, by Zend Technologies



Si tout s'affiche on supprime tout de suite ce fichier dangereux qui donne beaucoup trop d'informations sur notre serveur.

```
sudo rm /var/www/html/index.php
```

1.5 Activation de l'extension 'fileinfo'

GLPI va avoir besoin de l'extension PHP fileinfo. Il faut donc veiller à l'activer. Sur Debian cela est déjà fait, mais on peut le vérifier en tapant :

```
php -m | grep fileinfo
```

Sur Windows il va falloir l'activer dans le fichier de configuration de PHP `php.ini`, qui se trouve dans le dossier `php` de XAMPP.

On l'ouvre et on décommente la ligne correspondant à l'extension (faire une recherche de `fileinfo` dans le fichier).

Il suffit de supprimer le point-virgule au début de la ligne :

```
;extension=php_fileinfo.dll
```

Sur Debian il y aura d'autres lignes à modifier dans ce fichier, qui se trouve dans `/etc/php/7.0/apache2/php.ini`.

Les lignes suivantes doivent être adaptées :

```
memory_limit = 64M
file_uploads = on
max_execution_time = 600
session.auto_start = 0
session.use_trans_id = 0
```

Pour que tout changement soit pris en compte il faut recharger le serveur Apache2 :

— Sur Debian 9 :

```
sudo service apache2 restart
```

— Sur Windows :

Double cliquer sur `xampp-control` dans le répertoire de XAMPP et utiliser les boutons `Stop` puis `Start` dans la ligne de Apache.

1.6 Installation de GLPI

1.6.1 Préparation (Windows)

On extrait le fichier `glpi-0.90.tar.gz`. Cela nous donne un dossier `glpi` que l'on va placer en entier dans le répertoire `htdocs` de XAMPP.

1.6.2 Préparation (Debian)

La dernière version de GLPI n'étant pas compatible avec PHP < 7.2, nous allons prendre une plus ancienne version, c'est-à-dire la même que celle utilisée sur Windows pour cette procédure.

On va télécharger l'archive à l'adresse <https://github.com/glpi-project/glpi/releases/download/0.90.5/glpi-0.90.5.tar.gz>, puis la décompresser. On obtient un dossier `glpi` que l'on va placer en entier dans le répertoire `DocumentRoot` d'Apache : `/var/www/html/`.

Il faut maintenant donner les droits d'écriture sur deux répertoires de `glpi` :

```
— config/  
— files/  
sudo chmod -R 777 /var/www/html/glpi/{config,files}
```

1.6.3 Installation

On se rend dans un navigateur pour lancer l'installation proprement dite, grâce à la page <http://localhost/glpi/> :

- Choix de la langue.
- Accepter les termes de la licence.
- Choisir **Installer**.
- L'installateur effectue quelques tests, on va laisser les erreurs d'écriture. Tout en bas on peut cliquer sur **Continuer**.
- Se connecter à la base de données.
Pour le champ **Serveur MySQL** on met **localhost**.
 - Sur Windows par défaut il y a un utilisateur **root** avec un mot de passe vide.
 - Sur Debian on va utiliser les identifiants créés plus tôt : **glpi** avec le mot de passe **glpi**.
- Créer une base de données pour GLPI qu'on va appeler **glpi**.
- À la fin de l'installation on apprend que les identifiants administrateurs seront **glpi/glpi**.

Pour se connecter on se rendra à l'adresse <http://localhost/glpi/>.

1.7 Installation de Fusioninventory Server

1.7.1 Préparation (Windows)

On extrait le fichier `fusioninventory-for-glpi_0.90.1.4.tar.gz`. On place le dossier `fusioninventory` résultant dans le répertoire `plugins` de `glpi`.

1.7.2 Préparation (Debian)

On va télécharger la version `fusioninventory-for-glpi` pour notre version GLPI installée :

```
https://github.com/fusioninventory/fusioninventory-for-glpi/releases/download/glpi090+1.5/fusioninventory-for-glpi\_0.90.1.5.tar.gz
```

On décompresse et on place le dossier `fusioninventory` résultant dans le répertoire `plugins` de `glpi`.

1.7.3 Installation et activation

Pour installer et activer le plugin dans on se connecte sur <http://localhost/glpi/> (identifiants `glpi/glpi`) et on navigue à l'onglet **Configuration** > **Plugins**. On clique d'abord sur **Installer** puis sur **Activer**.

On finit par configurer l'URL d'accès au service. Pour cela aller dans le menu Administration > Entités > Root entity > Fusioninventory.

On ajoute `http://localhost/glpi` dans le champ prévu pour l'URL d'accès au service.

1.8 Installation de Fusioninventory Agent

1.8.1 sur Windows

Lancer `fusioninventory-agent_windows-x64_2.3.18.exe` et suivre l'installation. Il faut cocher tous les composants.

Dans la section serveur, mettre la liste des plugins à prendre en compte. Dans notre cas nous allons écrire `http://localhost/glpi/plugins/fusioninventory/`. On peut en ajouter d'autres, dans ce cas les séparer par des virgules.

1.8.2 sur Debian 9

Avant d'installer l'agent à proprement parler, il faut installer les dépendances :

```
sudo apt install dmidecode hwddata ucf hdparm
sudo apt install perl libuniversal-require-perl libwww-perl libparse-edid-perl
sudo apt install libproc-daemon-perl libfile-which-perl libhttp-daemon-perl
sudo apt install libxml-trepp-perl libyaml-perl libnet-cups-perl libnet-ip-perl
sudo apt install libdigest-sha-perl libsocket-getaddrinfo-perl libtext-template-perl
sudo apt install libxml-xpath-perl libyaml-tiny-perl
sudo apt install libnet-snmp-perl libcrypt-des-perl libnet-nbname-perl
sudo apt install libdigest-hmac-perl
sudo apt install libfile-copy-recursive-perl libparallel-forkmanager-perl
```

On va ensuite récupérer l'agent Fusioninventory sur le site internet. Il y a plusieurs fichiers `.deb` à télécharger pour que toutes les options soient actives :

```
https://github.com/fusioninventory/fusioninventory-agent/releases/download/2.5.2/fusioninventory-agent\_2.5.2-1\_all.deb
```

```
https://github.com/fusioninventory/fusioninventory-agent/releases/download/2.5.2/fusioninventory-agent-task-collect\_2.5.2-1\_all.deb
```

```
https://github.com/fusioninventory/fusioninventory-agent/releases/download/2.5.2/fusioninventory-agent-task-network\_2.5.2-1\_all.deb
```

```
https://github.com/fusioninventory/fusioninventory-agent/releases/download/2.5.2/fusioninventory-agent-task-deploy\_2.5.2-1\_all.deb
```

```
https://github.com/fusioninventory/fusioninventory-agent/releases/download/2.5.2/fusioninventory-agent-task-esx\_2.5.2-1\_all.deb
```

Pour les installer :

```
sudo dpkg -i fusioninventory-agent_2.5.2-1_all.deb
sudo dpkg -i fusioninventory-agent-task-collect_2.5.2-1_all.deb
sudo dpkg -i fusioninventory-agent-task-network_2.5.2-1_all.deb
sudo dpkg -i fusioninventory-agent-task-deploy_2.5.2-1_all.deb
sudo dpkg -i fusioninventory-agent-task-esx_2.5.2-1_all.deb
```

Il faut maintenant ouvrir le fichier `/etc/fusioninventory/agent.cfg` pour ajouter au moins une cible. On peut décommenter un exemple qui s'y trouve. On devrait avoir la ligne suivante :

```
server = http://localhost/glpi/plugins/fusioninventory/
```

On relance fusioninventory :

```
sudo systemctl restart fusioninventory-agent
```

1.9 Affichage de l'inventaire dans GLPI

Avant de pouvoir afficher quoi que ce soit sur GLPI il va falloir forcer un inventaire. On se rend sur la page de l'agent sur le port 62354 : `http://localhost:62354` et on clique sur *Force an inventory* :



```
This is FusionInventory Agent 2.3.18
The current status is running task Inventory
Force an Inventory
Next server target execution planned for:
```

- `http://192.168.26.137/glpi/plugins/fusioninventory: now`

Sur Debian on peut aussi envoyer au processus correspondant à l'agent un signal trap pour le forcer à exécuter l'inventaire :

```
sudo pkill -USR1 -f -P 1 fusioninventory-agent
```

On peut maintenant se connecter à l'interface web de GLPI pour visualiser l'inventaire offert par Fusioninventory. Dans un navigateur il faut se rendre à l'adresse `http://localhost/glpi/`. On entre l'identifiant `glpi` et le mot de passe `glpi`.

On peut ensuite se rendre dans l'onglet Parc > Ordinateurs et afficher l'inventaire :

GLPI

★ ⚙️ 🔌

Accueil > Parc > Ordinateurs
+ 🔍 ☰

+ +
Rechercher

Affichage (nombre d'éléments) 20

Page courante en PDF paysage 📄

Actions

	Nom	Statut	Fabricant	Numéro de série	Type	Modèle	Système d'exploitation	Lieu	Dernière modification	Composants - Processeur
<input type="checkbox"/>	debian9vm		innotek GmbH	2A628D65-1182-4EB6-9BB6-1B5FC072644D	VirtualBox	VirtualBox	Debian GNU/Linux 9 (stretch)		2020-10-21 15:45	Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz
<input type="checkbox"/>	Pc-formateur								2020-10-22 09:19	
<input type="checkbox"/>	WIN-LABO6-TSSR		LENOVO	S4KD8983	Desktop	10SJ50AV00	Microsoft Windows Server 2016 Standard		2020-10-22 08:52	Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz

Actions

Affichage (nombre d'éléments) 20

🔧

De 1 à 3 sur 3

0.018 seconde - 2.63 Mio
GLPI 0.90.5 Copyright (C) 2015 by Teclib - Copyright (C) 2003-2015 INDEPNET Development Team

2 Gestion du parc

2.1 Changement de l'intervalle de remontée

Par défaut l'agent Fusioninventory ne fait un inventaire que toutes les 24 heures. On va changer cette valeur pour faire un inventaire toutes les heures.

Dans l'interface de GLPI aller dans `Plugins > Fusioninventory > Général > Configuration générale` et changer la valeur de `Fréquence des inventaires (en heures)` à 1.

2.2 Ajout manuel d'une imprimante

Pour ajouter manuellement une imprimante on va dans `Parc > Imprimantes > +`. Puis on ajoute un nouveau gabarit vide.

Pour trouver les informations à mettre dans les champs, on a la chance d'avoir une imprimante qui possède une interface web. On va donc pouvoir se connecter directement dessus en visitant l'adresse IP `http://192.168.26.203`. On va changer les champs suivants :

- *Nom* : OKI-170FBF
- *Domaine* : WORKGROUP
- *Ports* : Ethernet
- *Type* : Noir et blanc
- *Fabricant* : OKI
- *Modèle* : B6200
- *Numéro de série* : 342110
- *Réseau* : 192.168.26.203

2.3 Administration de contrat

On va maintenant ajouter un contrat d'un an pour l'imprimante nouvellement créée.

`Gestion > Contrats > + > Gabarit vide`

- *Date de début* : 2020-01-01
- *Durée initiale du contrat* : 12 mois
- *Périodicité de facturation* : 1 mois

2.4 Remontée d'informations par SNMP

2.4.1 Préparation de l'imprimante

Il faut s'assurer que le SNMP est bien activé dans l'imprimante.

2.4.2 Préparation du switch

On se connecte en console au switch pour lui donner une adresse IP :

```
Switch#configure terminal
Switch(config)#interface vlan1
Switch(config-int)#ip address 192.168.26.38 255.255.255.0
Switch(config-int)#no shutdown
Switch(config-int)#end
```

et activer SNMP :

```
Switch#configure terminal
Switch(config)#snmp-server community public ro
Switch(config)#snmp-server enable traps snmp
Switch(config)#snmp-server contact "tssr5franken@gmail.com"
Switch(config)#snmp-server location "LAB06, AFPA Champs-sur-Marne"
Switch(config)#snmp-server host 192.168.26.137 version 2c public udp-port
162
Switch(config)#end
Switch#show snmp
```

... sans oublier de sauvegarder :

```
Switch#copy running-config startup-config
```

2.4.3 Ajout d'une plage IP

On doit ajouter à la main la plage IP pour les découvertes et les requêtes SNMP. Plugins > Fusioninventory > Réseau > Plage IP > +.

- dans Plage IP :
On va l'appeler IP LAB0 et on va lui donner 192.168.26.1 comme début et 192.168.26.254 comme fin.
- dans Authentification SNMP :
Associer les deux communautés v1 et v2c.

2.4.4 Activation de règles d'import

Pour éviter d'éventuels refus d'importation par certains périphériques lors du scan, il faut activer des règles d'import par adresse MAC.

Plugins > Fusioninventory > Règles > Règles d'import et de liaison des matériels.

On peut voir une liste de règles dont plusieurs pour les ordinateurs et les imprimantes. Dans les règles de mise à jour et dans les règles d'import, on constate que certaines règles par MAC sont inactives. En cliquant sur chaque règle concernée, sélectionner **Actif** > **Oui** et sauvegarder.

2.4.5 Ajout d'une tâche Découverte réseau

Plugins > Fusioninventory > Tâches > Gestion des tâches > +.

Dans le champ *Nom* on met `Découverte réseau`.

On peut ouvrir la tâche créée en cliquant dessus.

Dans l'onglet *Gestion des tâches* :

- cocher le champ `Actif`
- *Heure de démarrage programmée* : `2020-10-20 00:00`
- *Heure de fin programmée* : `2020-10-28 00:00`
- *Intervalle de réveil des agents (en minutes)* : `10`
- *Nombre d'agents à réveiller* : `1`

2.4.6 Ajout de plusieurs jobs pour la tâche

Maintenant qu'on a une tâche il va falloir créer un ou plusieurs jobs pour cette tâche.

Quand on ouvre la tâche, dans l'onglet *Configuration des jobs* :

- Ajouter un job
- *Nom* : `Découverte réseau`
- *Méthode du module* : `Découverte réseau`
- *Cibles* : `Plage IP > IP LABO`
- *Acteurs* : `Agent > debian9vm-2020-10-21-09-03-24`

2.4.7 Réduction des plages IP

Le scan d'une plage IP peut être très long, surtout une plage complète de classe C comme celle-ci. On peut donc pour les besoins de notre procédure ajouter plusieurs plages plus petites et ciblées dans le but d'y associer un job.

Créons par exemple une plage IP nommée `Imprimantes` démarrant à `192.168.26.203` et finissant à `192.168.26.204` de la même façon que tout à l'heure.

Puis Ajouter un job :

- *Nom* : `Découverte imprimantes`
- *Méthode du module* : `Découverte réseau`
- *Cibles* : `Plage IP > Imprimantes`
- *Acteurs* : `Agent > debian9vm-2020-10-21-09-03-24`

On peut faire de même pour d'autres équipements comme le switch par exemple.

Pour lancer le scan sur la plage IP cible, il faut forcer un inventaire sur l'agent.

3 Ticketing

Pour créer, gérer et suivre des tickets il faut créer des entités, des groupes, des comptes et des catégories de ticket.

3.1 Créer des entités

Aller dans **Administration > Entités**.

Nous avons par défaut une entité qui s'appelle **Root entity**. Nous allons créer trois entités additionnelles en cliquant sur +.

La première sera parent de la seconde qui sera parent de la troisième.

1. *Nom* : Île-de-France
Comme enfant de : Root entity
> Ajouter
2. *Nom* : Champs-sur-Marne
Comme enfant de : Île-de-France
> Ajouter
3. *Nom* : LAB06
Comme enfant de : Champs-sur-Marne
> Ajouter

3.2 Créer des groupes

Aller dans **Administration > Groupes**.

Un premier groupe est présent et s'appelle **WORKGROUP**. Nous allons en créer quatre de plus, toujours avec le bouton + au même endroit que précédemment :

1. *Nom* : Observateurs
Comme enfant de : ---
> Ajouter
2. *Nom* : Réseau
Comme enfant de : ---
> Ajouter
3. *Nom* : Système
Comme enfant de : ---
> Ajouter
4. *Nom* : Utilisateurs
Comme enfant de : ---
> Ajouter

3.3 Créer des comptes

Aller dans Administration > Utilisateurs.

Nous voyons les comptes créés lors de l'installation de GLPI. Nous allons maintenant créer deux comptes par groupe, toujours en utilisant le même + qu'avant. Nous allons aussi leur associer des Profils, qui sont déjà créés. L'entité et le profil sont ajoutés lors de la création de l'utilisateur. Le groupe en revanche est associé après coup. On commence donc par créer les Utilisateurs :

1. *Identifiant* : obs1
Profil : Observer
Entité : Île-de-France
> Ajouter
2. *Identifiant* : obs2
Profil : Observer
Entité : Île-de-France
> Ajouter
3. *Identifiant* : techR1
Profil : Technician
Entité : Champs-sur-Marne
> Ajouter
4. *Identifiant* : techR2
Profil : Technician
Entité : Champs-sur-Marne
> Ajouter
5. *Identifiant* : techS1
Profil : Technician
Entité : Champs-sur-Marne
> Ajouter
6. *Identifiant* : techS2
Profil : Technician
Entité : Champs-sur-Marne
> Ajouter
7. *Identifiant* : user1
Profil : Self-service
Entité : LAB06
> Ajouter
8. *Identifiant* : user2
Profil : Self-service
Entité : LAB06
> Ajouter

Maintenant que nos Utilisateurs sont créés, il faut leur associer un Groupe. Pour cela, effectuer pour chacun d'entre eux les actions suivantes :

- cocher l'Utilisateur
- cliquer sur **Actions**
- choisir **Associer à un groupe**
- choisir le groupe correspondant :
 - **Observateurs** pour **obs1** et **obs2**
 - **Réseau** pour **techR1** et **techR2**
 - **Système** pour **techS1** et **techS2**
 - **Utilisateurs** pour **user1** et **user2**

Il est évidemment possible de cocher les Utilisateurs deux par deux. . .

3.4 Créer des catégories de ticket

Aller dans **Configuration > Intitulés**. Dans la partie **Assistance** cliquer sur **Catégories de ticket**, puis sur **+** pour créer une nouvelle catégorie.

Nous allons créer deux catégories de tickets. Pour cela ne renseigner que le *Nom*, puis **> Ajouter**.

1. Réseau
2. Système

3.5 Créer des tickets

Maintenant que les entités, groupes, catégories et utilisateurs sont en place, nous pouvons créer des tickets.

Tout type de compte peut ouvrir un ticket. Aller dans **Assistance > Créer un ticket**. On peut maintenant entrer les informations nécessaires, mais tout n'est pas obligatoire. On retrouve, parmi d'autres, la **Catégorie** (**Réseau** ou **Système**), le **Demandeur** (avec le compte correspondant), l'**Observateur** et un champ **Attribué à** pour demander expressément à un compte particulier de résoudre le ticket.

Le champ **Description**, qui contient le corps du ticket, est le seul champ obligatoire, mais il vaut mieux ajouter au minimum un **Titre**.

Une fois le ticket créé, il apparaît dans la liste des **Tickets**.

3.6 Répondre à un ticket ou le résoudre

Si par exemple l'utilisateur **techR1** se connecte à son compte et souhaite afficher les tickets, il verra seulement les tickets qui lui sont attribués ou qu'il a écrits.

En cliquant sur le nom du ticket, il peut alors cliquer sur l'onglet **Traitement du ticket**. Il aura alors le choix entre :

- *Suivi* : une réponse dans la conversation du ticket, par exemple pour demander plus d'informations.
- *Tâche* : une réponse également, mais contenant une directive pour le demandeur.
- *Document* : pour simplement ajouter un document en pièce jointe.
- *Solution* : pour apporter une solution sensée clore le ticket. Dans ce cas le demandeur aura la possibilité soit de valider la solution pour clore le ticket, soit de refuser la solution si elle ne marche pas, ce qui aura pour effet de maintenir la conversation ouverte.

4 Sources

<https://docs.ovh.com/fr/dedicated/installer-lamp-debian-ubuntu/>

<https://glpi-install.readthedocs.io/en/latest>

<https://fusioninventory.org/get/>

<https://techbast.com/2020/06/how-to-configure-monitoring-switch-cisco-with-snmp-protocol-on-zabbix.html>