

# Compétence C1

## Recensement et identification des ressources numériques

---

<b>Compétence</b>	C1 — Recenser et identifier les ressources numériques
<b>Formation</b>	BTS SIO option SISR — 1ère année — EFREI
<b>Outil principal</b>	GLPI 10.0.20
<b>Environnement</b>	VMware Workstation, Debian GNU/Linux 13.3.0 (amd64)
<b>Adresse du serveur</b>	<a href="http://192.168.10.128">http://192.168.10.128</a>
<b>Date</b>	Avril 2026

Ce document retrace les étapes de mise en place d'un système d'inventaire automatisé du parc informatique dans le cadre de la compétence C1 du BTS SIO option SISR. L'outil utilisé est GLPI (Gestion Libre de Parc Informatique), déployé sur une machine virtuelle Debian sous VMware Workstation. Une fois l'installation terminée, des agents GLPI sont déployés sur deux machines distinctes afin de remonter automatiquement leurs informations matérielles et logicielles vers le serveur central. Les données collectées sont ensuite consultables et exportables depuis l'interface web.

# 1. Environnement technique

Une machine virtuelle est créée sous VMware Workstation à partir de l'image ISO **Debian 13.3.0 netinst (amd64)**. Cette VM héberge l'ensemble de la pile applicative (serveur web, base de données, GLPI). Le mode réseau **NAT** permet à la machine virtuelle d'accéder à Internet via l'adresse IP de la machine hôte tout en restant joignable depuis le navigateur de l'hôte via son adresse IP locale.

Composant	Configuration
Système d'exploitation	Debian GNU/Linux 13.3.0 (amd64)
RAM	4 Go (4096 Mo)
Processeur	1 socket / 2 cœurs
Disque	20 Go (SCSI, allocation dynamique)
Réseau	NAT

## 1.1 Configuration VMware

Les captures suivantes présentent les paramètres de la machine virtuelle tels qu'ils ont été configurés dans VMware Workstation avant le premier démarrage du système.

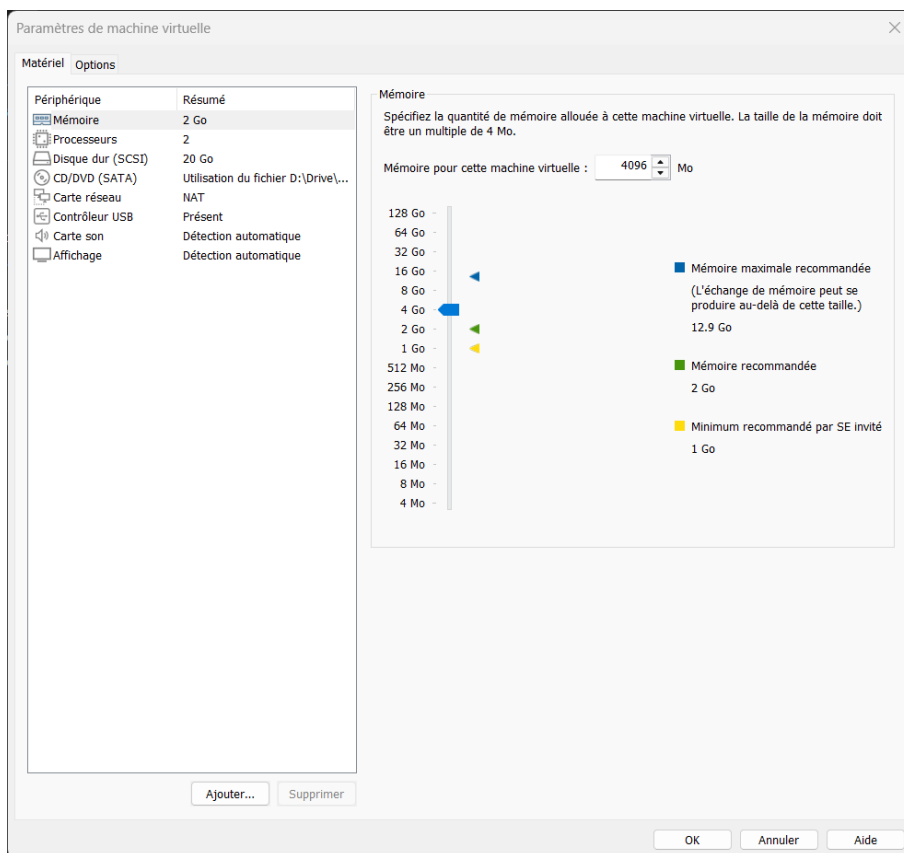


Figure 1 — Paramètres VMware : RAM allouée (4096 Mo)

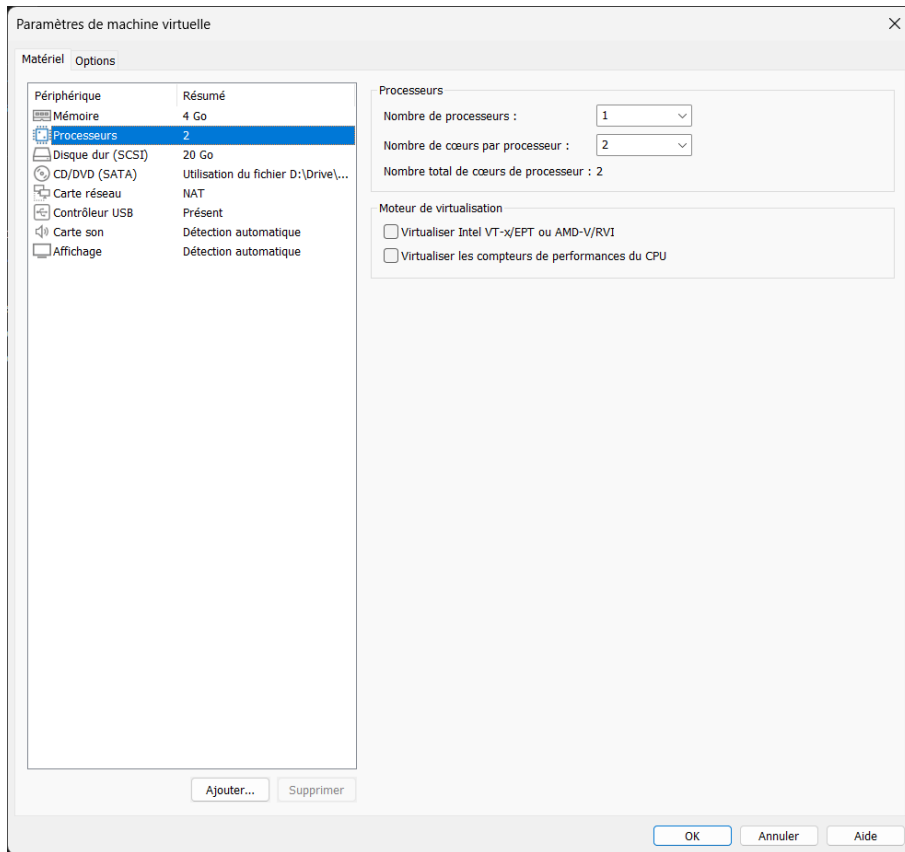


Figure 2 — Paramètres VMware : processeur (1 socket / 2 cœurs)

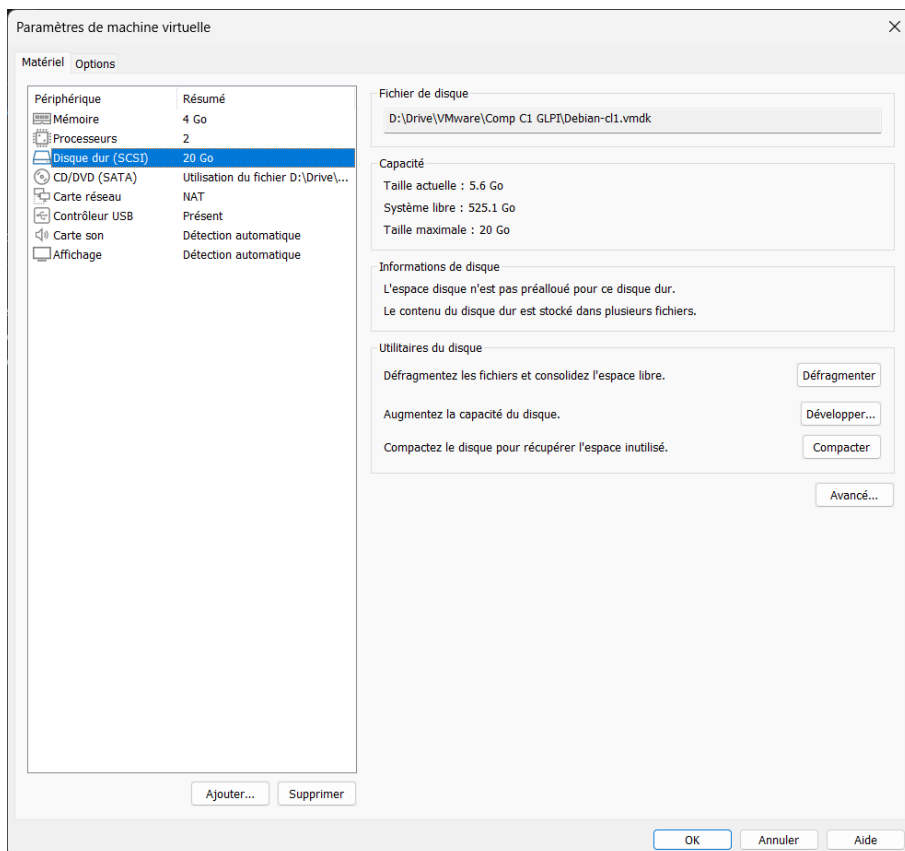


Figure 3 — Paramètres VMware : disque dur (20 Go, SCSI)

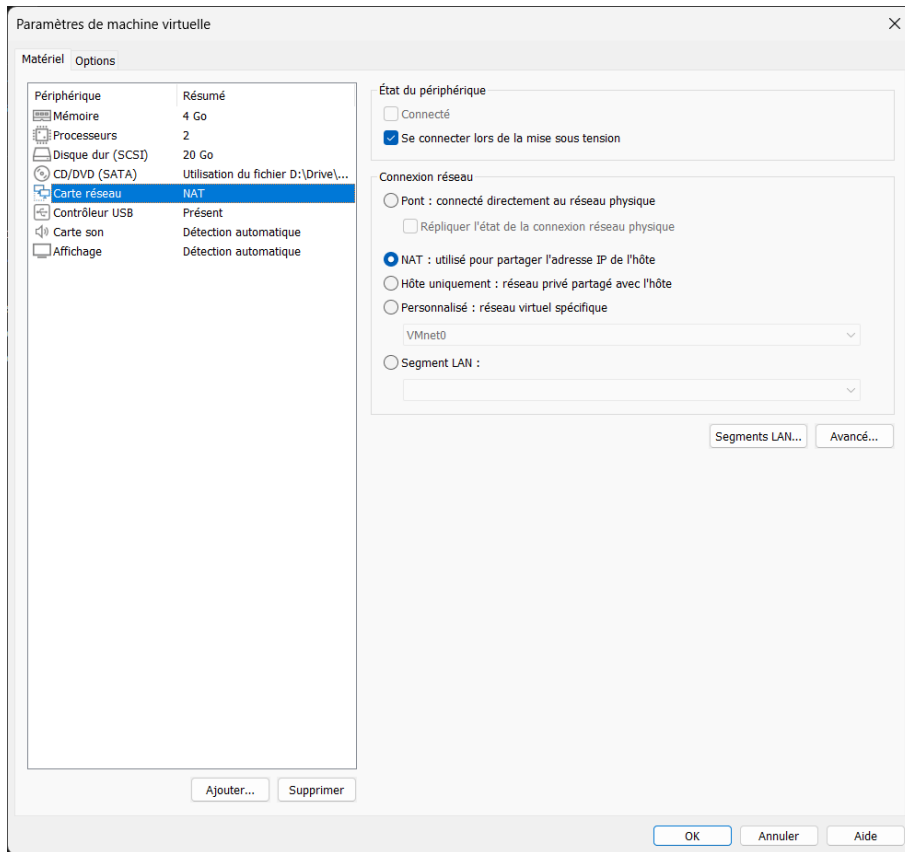


Figure 4 — Paramètres VMware : carte réseau en mode NAT

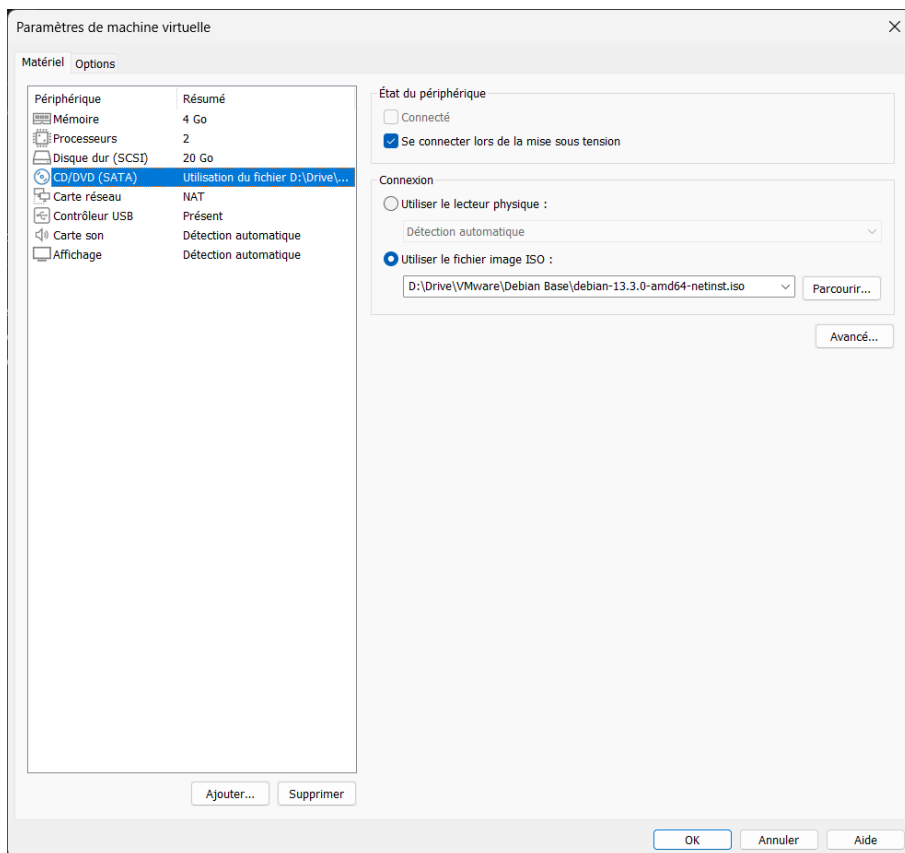


Figure 5 — Paramètres VMware : image ISO Debian 13.3.0 sélectionnée

## 2. Installation de GLPI

GLPI s'appuie sur une pile applicative **LAMP** (Linux, Apache, MariaDB, PHP). Chaque composant est installé séparément avant de procéder au déploiement de GLPI via son assistant web.

Composant	Rôle
Linux (Debian 13)	Système d'exploitation hôte
Apache 2.4	Serveur web qui sert l'interface GLPI
MariaDB	Base de données stockant toutes les informations de GLPI
PHP 8.3	Interpréteur exécutant le code de GLPI

### 2.1 Mise à jour du système

Première étape : passer en utilisateur **root** et synchroniser la liste des paquets disponibles dans les dépôts officiels.

```
su
apt update
```

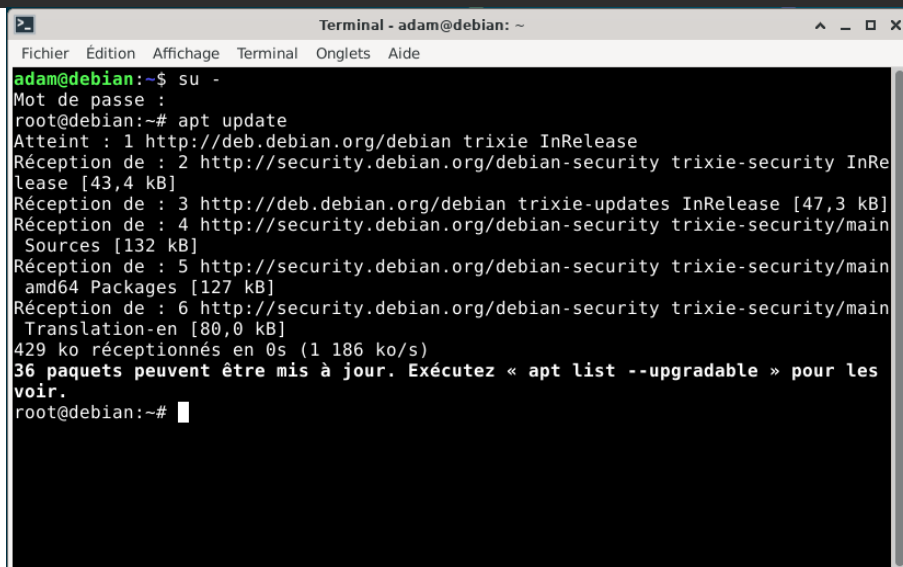


Figure 6 — Terminal : résultat de apt update sur Debian 13

### 2.2 Installation d'Apache et récupération de l'adresse IP

Le serveur web Apache est installé, puis l'adresse IP de la VM est relevée afin de pouvoir accéder à l'application depuis le navigateur de la machine hôte.

```
apt install apache2
ip address show
```

```
Applications : Terminal - adam@debia...
Terminal - adam@debian: ~
Fichier Edition Affichage Terminal Onglets Aide
Paramétrage de apache2-utils (2.4.66-1-deb13u2) ...
Paramétrage de apache2-bin (2.4.66-1-deb13u2) ...
Paramétrage de apache2 (2.4.66-1-deb13u2) ...
Enabling module mpm_event.
Enabling module authz_core.
Enabling module authz_host.
Enabling module authn_core.
Enabling module authn_basic.
Enabling module access_compat.
Enabling module authn_file.
Enabling module authz_user.
Enabling module alias.
Enabling module dir.
Enabling module autoindex.
Enabling module env.
Enabling module mime.
Enabling module negotiation.
Enabling module setenvif.
Enabling module filter.
Enabling module deflate.
Enabling module status.
Enabling module reqtimeout.
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /usr/lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /usr/lib/systemd/system/apache-htcacheclean.service.
Traitement des actions différées (« triggers ») pour man-db (2.13.1-1) ...
Traitement des actions différées (« triggers ») pour libc-bin (2.41-12+deb13u2) ...
root@debian:~# ip address show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1::1 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:92:1b:f1 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    altname emx800c29921bf1
    inet 192.168.10.128/24 brd 192.168.10.255 scope global dynamic noprefixroute ens33
        valid_lft 1597sec preferred_lft 1597sec
    inet6 fe80::20c:29ff:fe92:1bf1/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
root@debian:~#
```

Figure 7 — Terminal : installation d'Apache et adresse IP de la VM (192.168.10.128)

## 2.3 Vérification du serveur web

Depuis le navigateur de la machine hôte, l'accès à l'adresse IP de la VM (<http://192.168.10.128>) doit afficher la page d'accueil par défaut d'Apache, confirmant ainsi le bon fonctionnement du serveur web.

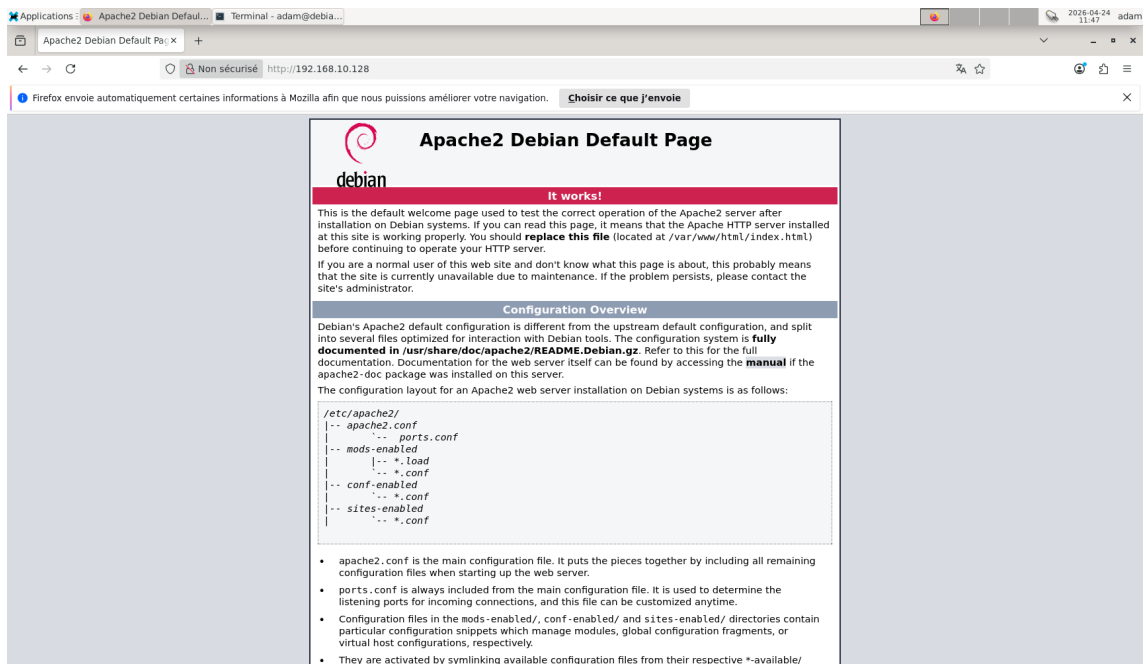


Figure 8 — Navigateur : page par défaut d'Apache confirmant le bon fonctionnement du serveur

## 2.4 Installation de PHP 8.3

Debian 13 ne fournit pas PHP 8.3 dans ses dépôts officiels par défaut. Il faut donc ajouter le dépôt tiers de **Sury**, qui maintient les dernières versions de PHP pour Debian, puis installer PHP ainsi que les extensions requises par GLPI.

```
sudo apt install -y lsb-release ca-certificates apt-transport-https curl
sudo curl -sSLo /tmp/debsuryorg-archive-keyring.deb \
```

```

https://packages.sury.org/debsuryorg-archive-keyring.deb
sudo dpkg -i /tmp/debsuryorg-archive-keyring.deb
sudo apt update && sudo apt install php8.3
sudo apt install php8.3-xml php8.3-dom php8.3-simplexml php8.3-curl \
php8.3-gd php8.3-intl php8.3-mbstring php8.3-zip php8.3-mysqli
sudo apt install php8.3-bz2 php8.3-phar php8.3-exif php8.3-ldap \
openssl php8.3-opcache php8.3-readline

```

```

Préparation du dépaquetage de .../7-php8.3_8.3.30-3+0-20260213.77+debian13-1.gbpa1591f_all.deb ...
Dépaquetage de php8.3 (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
Paramétrage de psmisc (23.7-2) ...
Paramétrage de php-common (2:101+0-20260212.63+debian13-1.gbp7d9167) ...
Created symlink '/etc/systemd/system/timers.target.wants/phpsessionclean.timer' → '/usr/lib/systemd/system/phpsessionclean.timer'.
Paramétrage de php8.3-common (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
Creating config file /etc/php/8.3/mods-available/calendar.ini with new version
Creating config file /etc/php/8.3/mods-available/ctype.ini with new version
Creating config file /etc/php/8.3/mods-available/exif.ini with new version
Creating config file /etc/php/8.3/mods-available/fileinfo.ini with new version
Creating config file /etc/php/8.3/mods-available/ffi.ini with new version
Creating config file /etc/php/8.3/mods-available/ftp.ini with new version
Creating config file /etc/php/8.3/mods-available/gettext.ini with new version
Creating config file /etc/php/8.3/mods-available/iconv.ini with new version
Creating config file /etc/php/8.3/mods-available/pdo.ini with new version
Creating config file /etc/php/8.3/mods-available/phar.ini with new version
Creating config file /etc/php/8.3/mods-available/posix.ini with new version
Creating config file /etc/php/8.3/mods-available/shmop.ini with new version
Creating config file /etc/php/8.3/mods-available/sockets.ini with new version
Creating config file /etc/php/8.3/mods-available/sysvmsg.ini with new version
Creating config file /etc/php/8.3/mods-available/syssem.ini with new version
Creating config file /etc/php/8.3/mods-available/sysvshm.ini with new version
Creating config file /etc/php/8.3/mods-available/tokenizer.ini with new version
Paramétrage de php8.3-readline (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
Paramétrage de php8.3-opcache (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
Creating config file /etc/php/8.3/mods-available/opcache.ini with new version
Paramétrage de php8.3-cli (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
update-alternatives: utilisation de « /usr/bin/php8.3 » pour fournir « /usr/bin/php » (php) en mode automatique
update-alternatives: utilisation de « /usr/bin/phar8.3 » pour fournir « /usr/bin/phar » (phar) en mode automatique
update-alternatives: utilisation de « /usr/bin/phar.phar8.3 » pour fournir « /usr/bin/phar.phar » (phar.phar) en mode automatique
Creating config file /etc/php/8.3/cli/php.ini with new version
Paramétrage de libapache2-mod-php8.3 (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
Creating config file /etc/php/8.3/apache2/php.ini with new version
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
apache2_invoke: Enable module php8.3
Paramétrage de php8.3 (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
Traitement des actions différées (« triggers ») pour man-db (2.13.1-1) ...
Traitement des actions différées (« triggers ») pour php8.3-cli (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
Traitement des actions différées (« triggers ») pour libapache2-mod-php8.3 (8.3.30-3+0-20260213.77+debian13-1.gbpa1591f) ...
root@debian:~# php -v
PHP 8.3.30 (cli) (built: Feb 13 2026 15:52:05) (NTS)
Copyright (c) The PHP Group
Zend Engine v4.3.30, Copyright (c) Zend Technologies
with Zend OPcache v8.3.30, Copyright (c), by Zend Technologies
root@debian:~#

```

Figure 9 — Terminal : PHP 8.3.30 correctement installé (php -v)

## 2.5 Installation de MariaDB

MariaDB est le système de gestion de base de données utilisé par GLPI pour stocker l'ensemble de ses données (équipements, utilisateurs, tickets, configuration, etc.). C'est un fork libre et compatible de MySQL.

```

apt install mariadb-server
systemctl restart apache2

```

## 2.6 Téléchargement et déploiement de GLPI

L'archive GLPI est téléchargée depuis le dépôt officiel sur GitHub, décompressée, puis déplacée dans le répertoire de service web **/var/www/** où Apache pourra la trouver.

```

sudo wget https://github.com/glpi-project/glpi/releases/download/10.0.20/glpi-10.0.20.tgz
tar -xvzf glpi-10.0.20.tgz
mv glpi /var/www/

```

## 2.7 Configuration du virtual host Apache

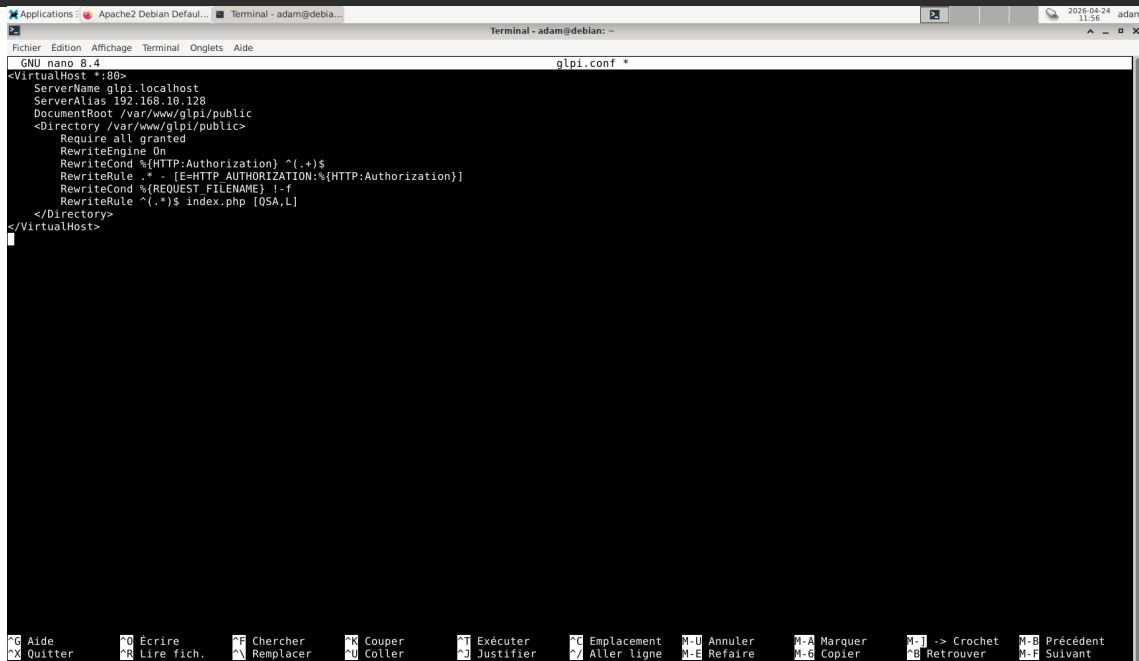
Un **virtual host** permet à Apache de savoir quel répertoire servir lorsqu'il reçoit une requête sur l'adresse IP de la VM. Le module **rewrite** est préalablement activé pour permettre la gestion des URLs propres de GLPI.

```

sudo a2enmod rewrite
cd /etc/apache2/sites-available

```

nano glpi.conf



```
GNU nano 8.4 glpi.conf *
<VirtualHost *:80>
  ServerName glpi.localhost
  ServerAlias 192.168.10.128
  DocumentRoot /var/www/glpi/public
  <Directory /var/www/glpi/public>
    Require all granted
    RewriteEngine On
    RewriteCond %{HTTP:Authorization} ^(.+)$
    RewriteRule .* - [E=HTTP_AUTHORIZATION:%{HTTP:Authorization}]
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteRule ^(.*)$ index.php [QSA,L]
  </Directory>
</VirtualHost>
```

Figure 10 — Fichier glpi.conf édité dans nano avec la configuration du virtual host

Une fois le fichier créé, le site est activé et les droits du dossier GLPI sont réattribués à l'utilisateur **www-data** (utilisateur sous lequel Apache s'exécute), afin qu'il puisse écrire dans les répertoires nécessaires.

```
sudo a2ensite glpi.conf
sudo chown www-data:www-data /var/www/glpi/config/
sudo chown -R www-data:www-data /var/www/glpi/files/
sudo chown www-data:www-data /var/www/glpi/marketplace
```

## 2.8 Sécurisation des cookies PHP et configuration de MariaDB

Dans `/etc/php/8.3/apache2/php.ini`, la directive `session.cookie_httponly` est passée à **On** afin d'empêcher les scripts JavaScript côté client d'accéder aux cookies de session, ce qui constitue une protection contre les attaques de type XSS. Ensuite, le mot de passe root de MariaDB est défini.

```
sudo nano /etc/php/8.3/apache2/php.ini
# Modifier : session.cookie_httponly = On
systemctl reload apache2

mariadb
ALTER USER 'root'@'localhost' IDENTIFIED BY '[MOT_DE_PASSE]';
FLUSH PRIVILEGES;
exit
```

## 2.9 Installation via l'interface web

Depuis le navigateur, l'accès à l'IP de la VM lance l'assistant d'installation GLPI. Après vérification des prérequis (qui doivent tous être au vert), la connexion à la base de données est configurée avec l'hôte **127.0.0.1** et les identifiants MariaDB définis précédemment. Une fois l'assistant terminé, l'application affiche sa page de connexion.

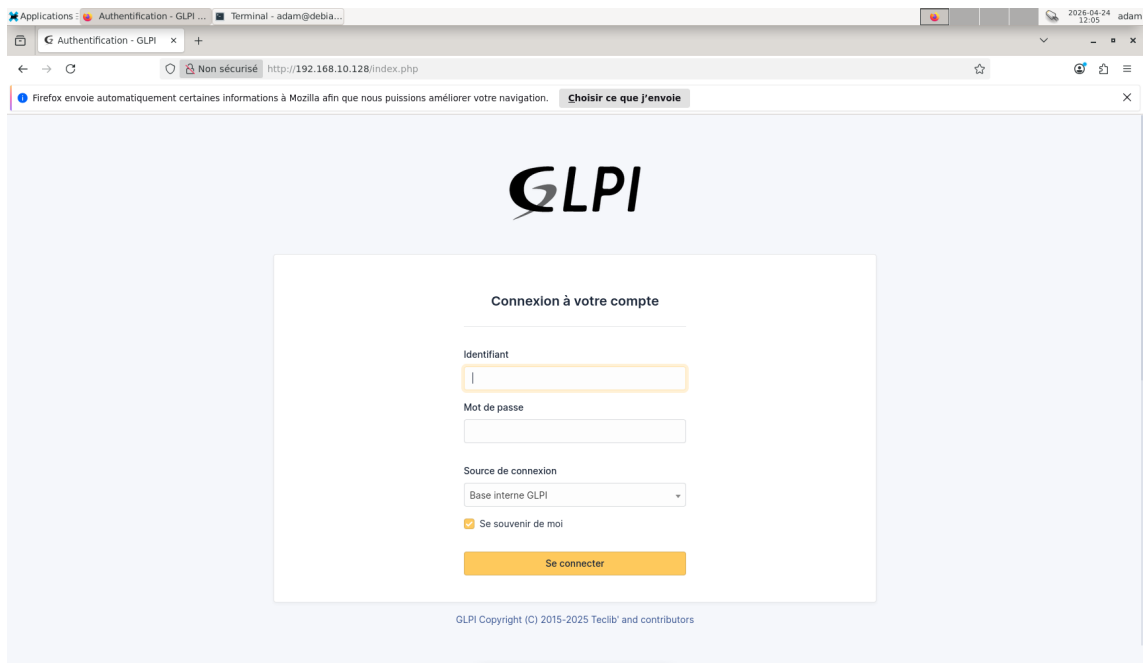


Figure 11 — Page de connexion GLPI après installation complète

La connexion avec le compte par défaut **glpi / glpi** donne accès au tableau de bord du Super-Admin, depuis lequel toute la configuration de l'application sera réalisée.

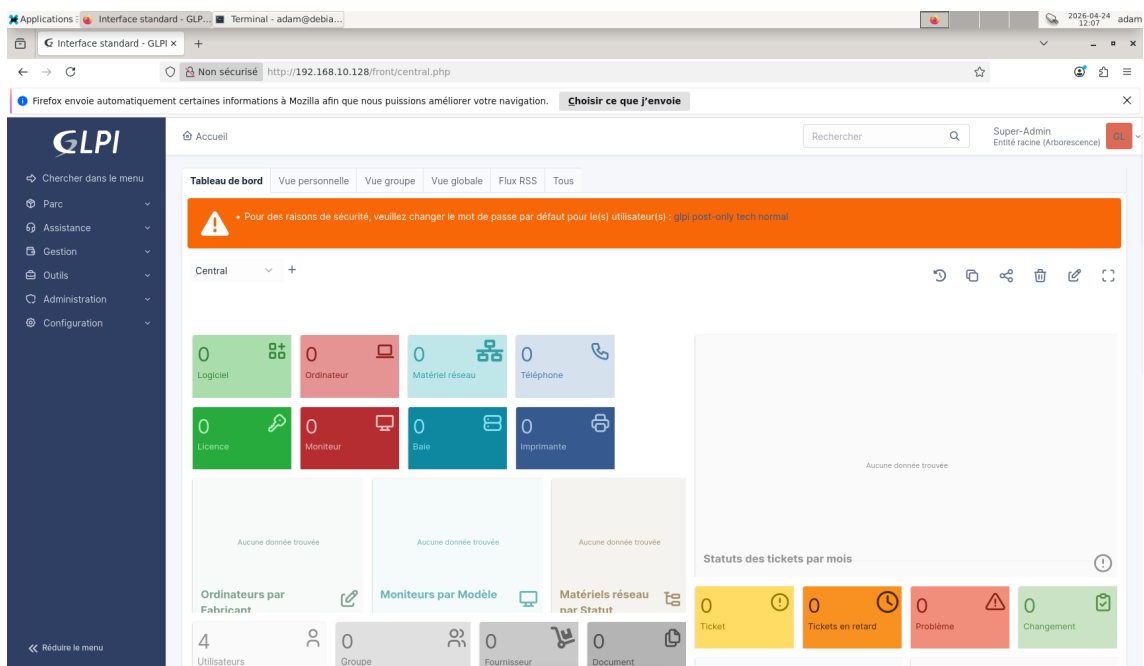


Figure 12 — Tableau de bord GLPI connecté en Super-Admin

### 3. Activation de l'inventaire automatisé

Pour que GLPI puisse recevoir les inventaires des agents installés sur les machines du parc, le plugin **GLPI Inventory** doit être installé depuis la marketplace intégrée. Une clé d'activation gratuite est nécessaire, obtenue depuis un compte sur [glpi-project.org](http://glpi-project.org).

### 3.1 Installation du plugin GLPI Inventory

Depuis **Configuration** → **Marketplace**, après avoir renseigné la clé d'enregistrement, le plugin **GLPI Inventory** est recherché, installé puis activé. Le bandeau d'information confirme l'activation réussie.

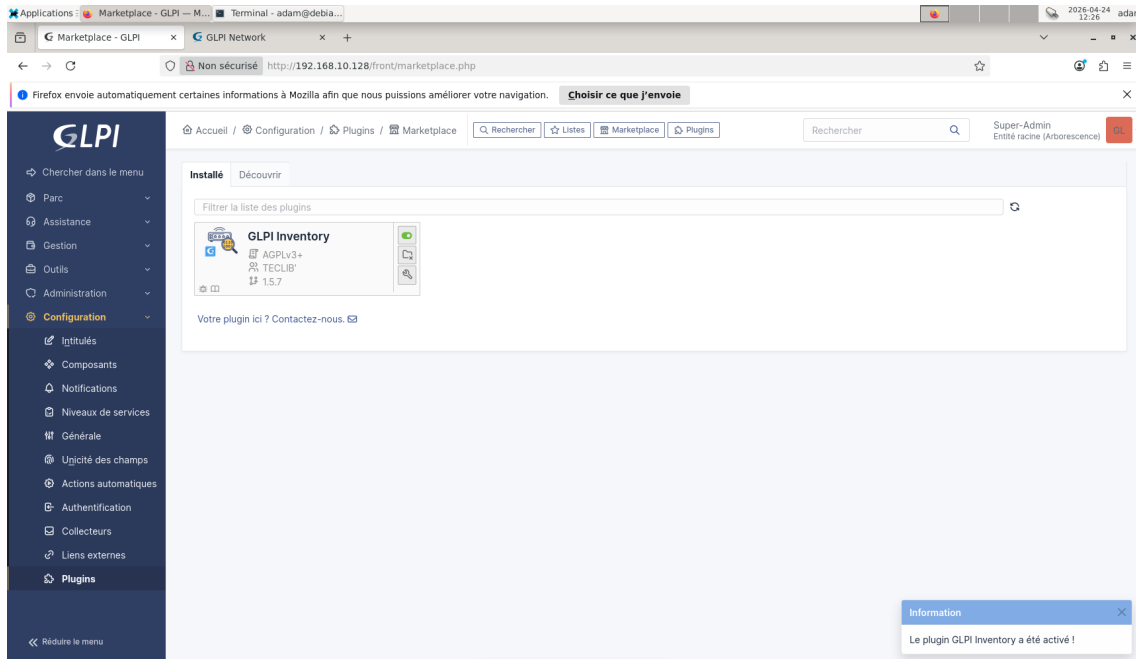


Figure 13 — Marketplace GLPI : plugin GLPI Inventory installé et activé (v1.5.7)

### 3.2 Configuration de la collecte

Dans **Administration** → **Inventaire**, l'option **Activer l'inventaire** est cochée pour autoriser la réception des données entrantes. Les options d'importation sont activées sur l'ensemble des éléments collectés : volumes, logiciels, moniteurs, périphériques, processeurs, disques, etc. Cette configuration permet d'obtenir l'inventaire le plus complet possible des machines du parc.

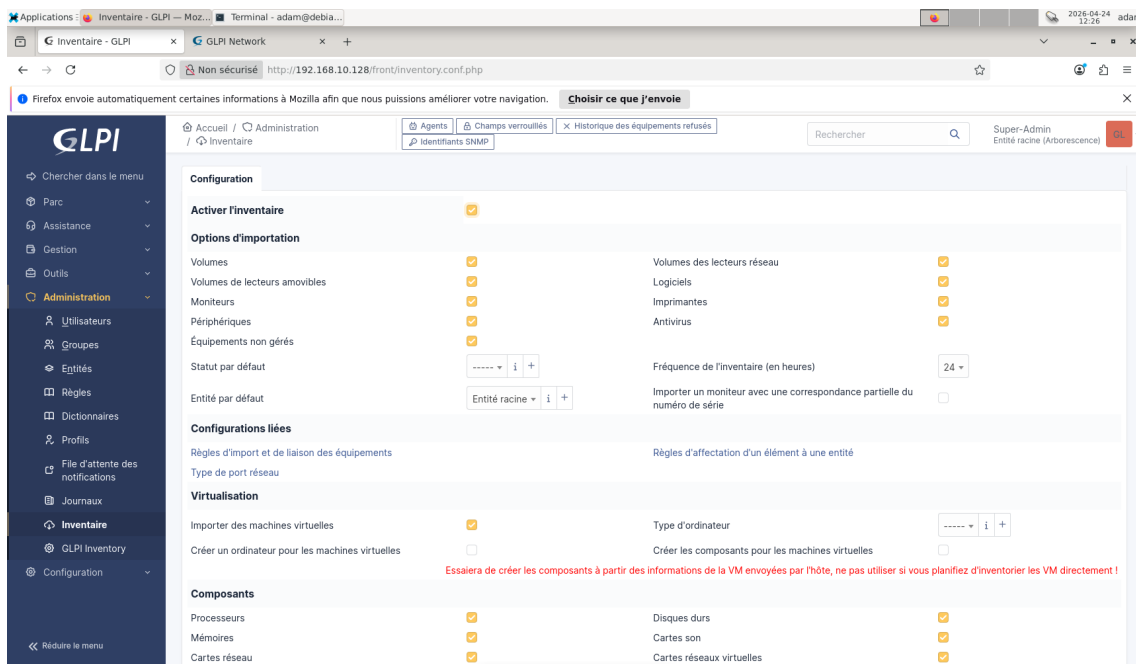


Figure 14 — GLPI Administration > Inventaire : collecte activée avec toutes les options d'importation

## 4. Installation de l'agent GLPI sur le serveur

L'agent GLPI est un programme qui doit être installé sur chaque machine à inventorier. Il collecte automatiquement les informations système (matériel, logiciels installés, configuration réseau, etc.) puis les envoie au serveur GLPI via une requête HTTP. Pour cette première mise en pratique, l'agent est installé directement sur la VM serveur afin qu'elle s'inventorie elle-même.

```
wget https://github.com/glpi-project/glpi-agent/releases/download/1.15/glpi-agent-1.15-linux-installer.pl
sudo perl glpi-agent-1.15-linux-installer.pl

# Lancement de l'inventaire
sudo glpi-agent --server http://192.168.10.128/front/inventory.php
```

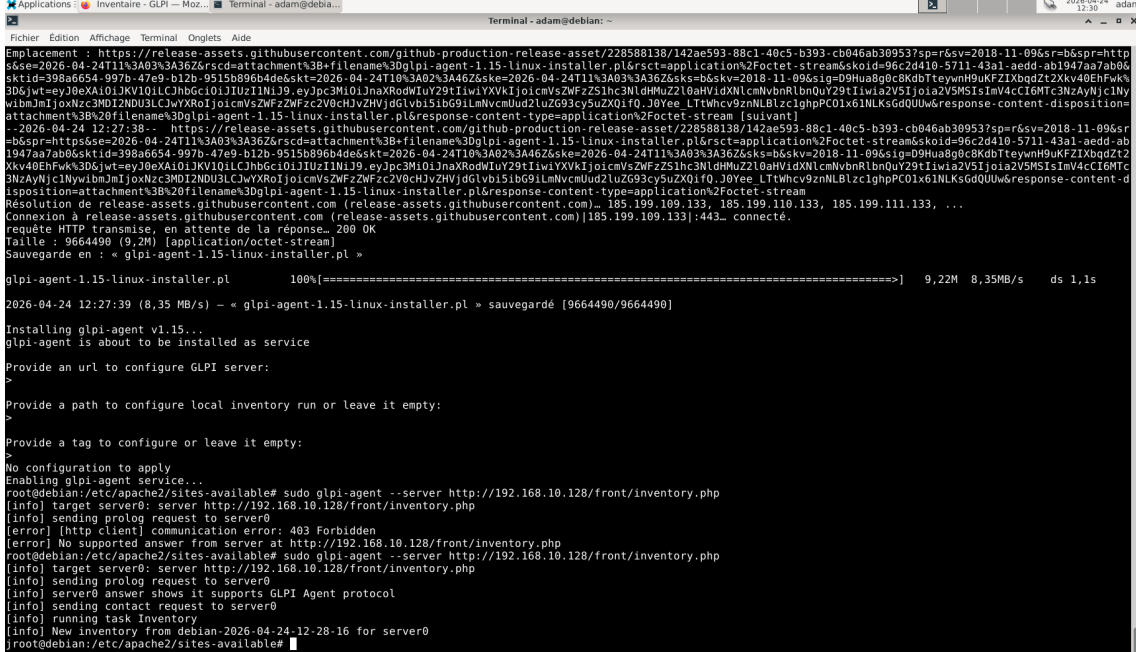


Figure 15 — Terminal : installation de l'agent GLPI v1.15 et envoi de l'inventaire

Une fois l'inventaire envoyé, la machine apparaît automatiquement dans **Parc** → **Ordinateurs** avec l'ensemble de ses caractéristiques détectées : fabricant, numéro de série, système d'exploitation, processeur, etc.

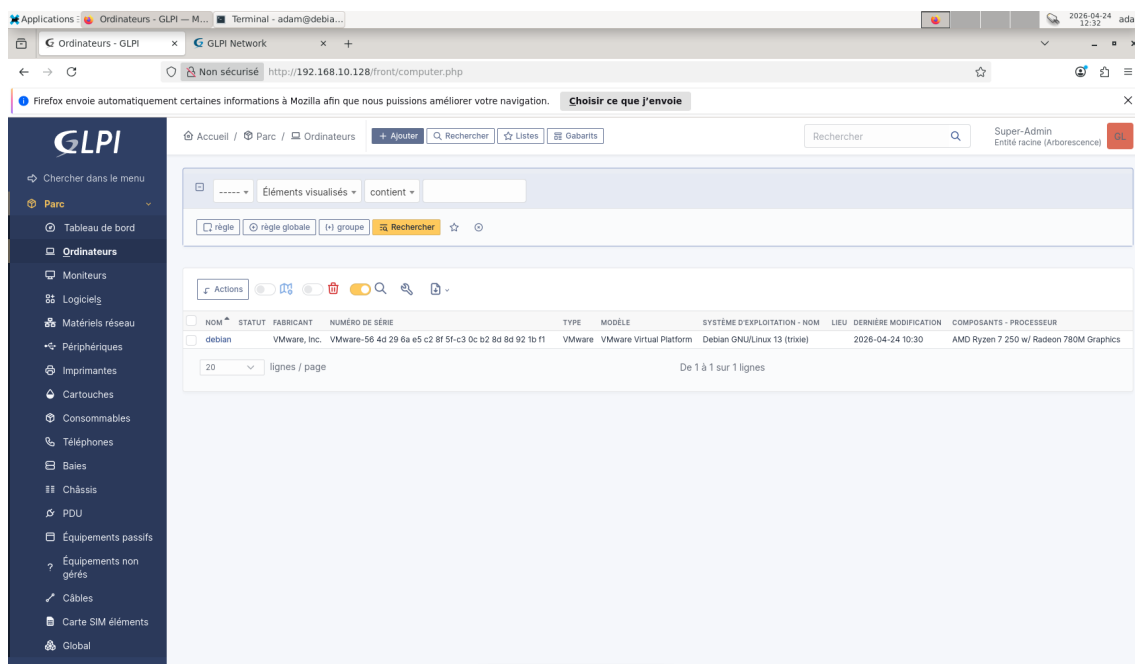


Figure 16 — GLPI Parc > Ordinateurs : le serveur debian apparaît dans l'inventaire

## 5. Inventaire d'une seconde machine

Pour démontrer le caractère **centralisé** de l'inventaire, une seconde machine virtuelle Debian est créée puis configurée. L'agent GLPI y est installé et pointé vers le serveur GLPI déjà en place. Ses caractéristiques principales sont identiques à la première VM, seule l'adresse IP diffère.

Paramètre	Valeur
Nom de la machine	debian (VM secondaire)
Adresse IP	192.168.10.135
Système d'exploitation	Debian GNU/Linux 13.3.0 (amd64)
Serveur GLPI cible	http://192.168.10.128/front/inventory.php



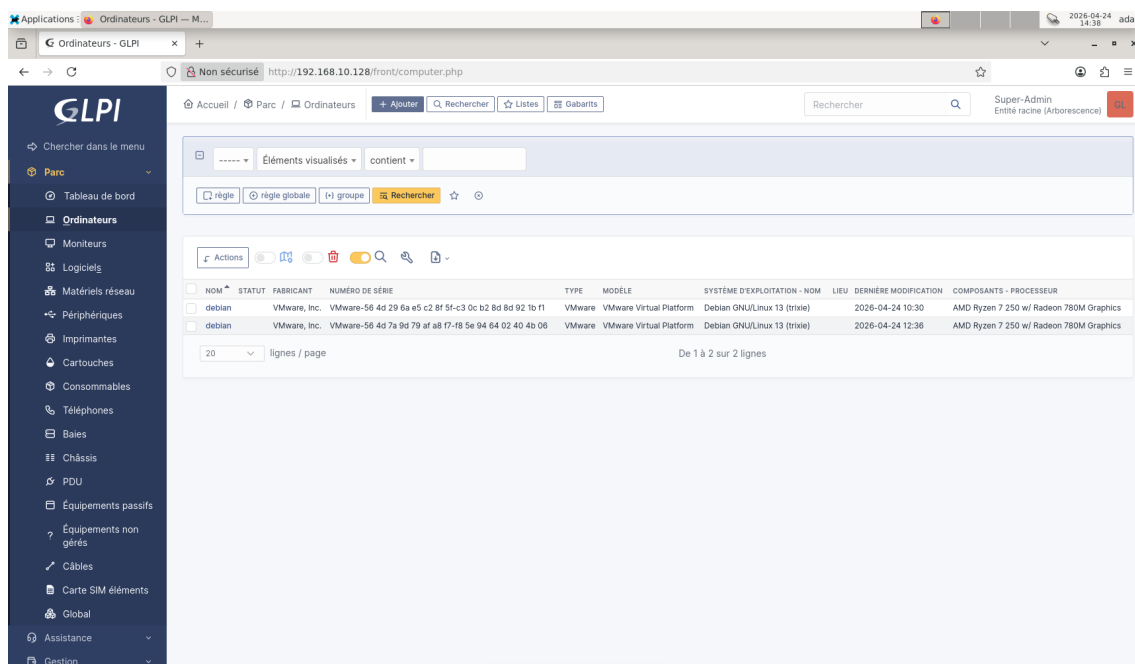


Figure 19 — GLPI Parc > Ordinateurs : les deux machines debian inventoriées (2 sur 2 lignes)

## 6. Export des données d'inventaire

Une fois les machines inventoriées, GLPI permet d'exporter les données collectées aux formats **CSV** ou **PDF** via le bouton d'export disponible en haut de chaque liste. Les exports sont accessibles depuis **Parc** → **Ordinateurs** pour les machines, et **Parc** → **Logiciels** pour les applications détectées.

Nom	Statut	Fabricant	Numéro de série	Type	Modèle	Système d'exploitation - Nom	Lieu	Dernière modification	Composants - Processeur
debian		VMware, Inc.	VMware-56 4d 29 6a e5 c2 8f 5f-c3 0c b2 8d 8d 92 1b f1	VMware	VMware Virtual Platform	Debian GNU/Linux 13 (trixie)		2026-04-24 10:30	AMD Ryzen 7 250 w/ Radeon 780M Graphics
debian		VMware, Inc.	VMware-56 4d 7a 9d 79 af a8 f7-f8 5e 94 64 02 40 4b 06	VMware	VMware Virtual Platform	Debian GNU/Linux 13 (trixie)		2026-04-24 12:36	AMD Ryzen 7 250 w/ Radeon 780M Graphics

Figure 20 — Export CSV de la liste des ordinateurs : deux machines debian recensées

L'export contient les informations principales de chaque machine : nom, fabricant, numéro de série, type, modèle, système d'exploitation, date de dernière modification et processeur détecté. Ces données peuvent être exploitées pour produire des rapports d'audit, des inventaires comptables ou des fichiers de référence pour des projets de migration.

## Bilan

À l'issue de ces manipulations, le parc informatique simulé, composé de deux machines virtuelles Debian, est entièrement recensé dans GLPI. Chaque machine remonte automatiquement son inventaire matériel et logiciel via l'agent GLPI, et les données sont consultables et exportables depuis l'interface web centralisée.

Cette mise en pratique illustre concrètement le processus d'identification des ressources numériques tel qu'il est mis en œuvre en entreprise : déploiement d'un serveur de gestion de parc, installation d'agents sur les postes à inventorier, collecte centralisée des données et exploitation via des exports. L'avantage

principal de cette approche est l'**automatisation** : il n'est plus nécessaire de saisir manuellement les caractéristiques de chaque machine, et toute évolution (changement de matériel, ajout de logiciels) est détectée lors du prochain envoi d'inventaire.

Tâche	Statut
Création de la VM serveur sous VMware Workstation	Réalisé
Installation de la pile LAMP (Apache, MariaDB, PHP 8.3)	Réalisé
Installation de GLPI 10.0.20 sur le serveur	Réalisé
Activation du plugin GLPI Inventory	Réalisé
Configuration de la collecte d'inventaire	Réalisé
Installation de l'agent GLPI sur le serveur (auto-inventaire)	Réalisé
Création d'une seconde VM Debian	Réalisé
Installation de l'agent GLPI sur la VM2	Réalisé
Envoi de l'inventaire de la VM2 vers le serveur central	Réalisé
Export des ordinateurs au format CSV	Réalisé