

DOCUMENTATION TECHNIQUE

Installation et Configuration du GLPI sur debian

Agence Immobilière Rue25

Debian & GLPI

Debian

Version 1.0 - Novembre 2025

Le Sommaire

Contexte.....	3
Étape 1 création de la machine virtuelle :.....	4
Installation non graphique de Debian :.....	6
Étape 2 : Accès au ROOT et mise à jour du système :.....	12
Étape 3 Installation du serveur SSH.....	14
Étape 4 configuration réseau statique :.....	16
Étape 5 Installation du serveur	20
Étape 6 Sécurisation de MariaDB.....	23
Étape 7 Création de la base GLPI :.....	24
Étape 8 téléchargement et déploiement de GLPI.....	27
Étape 9 Configuration Apache.....	29
Étape 10 Installation via l'interface web.....	31
Sources :.....	35
Installation complet GLPI 10 sur Debian.....	35
Gestion des utilisateurs, profils et habilitation dans GLPI.....	35
Test réseau : ping, SSH, connectivité.....	35

Contexte

Dans le cadre du déploiement du parc informatique de l'agence rue25, il a été décidé de mettre en place une solution de gestion centralisée des équipements et des demandes d'assistance technique. Le choix s'est porté sur GLPI (Gestionnaire Libre de Parc Informatique), un outil open source reconnu pour sa robustesse, sa modularité et sa capacité à structurer efficacement les processus ITSM (IT service Management).

Ce rapport technique documente l'intégralité du processus d'installation, de configuration et de sécurisation de GLPI sur une machine virtuelle Debian 11.6 avec interface graphique GNOM. Il s'inscrit dans une démarche professionnelle visant à :

- Déployer une infrastructure fiable et sécurisée
- Automatiser la gestion du parc informatique
- Centraliser les demandes de support
- Assurer la traçabilité des interventions
- Préparer un environnement évolutif pour l'agence rue25

L'installation a été réalisée dans un environnement virtualisé, avec des ressources limitées (1CPU, 2Go RAM, 20 Go disque) ; afin de simuler un contexte réaliste de déploiement en PME. Le serveur Debian héberge les services Apache, MariaDB et PHP nécessaire au fonctionnement de GLPI. Il est configuré pour être accessible sur le réseau local via une IP statique.

Ce document présente :

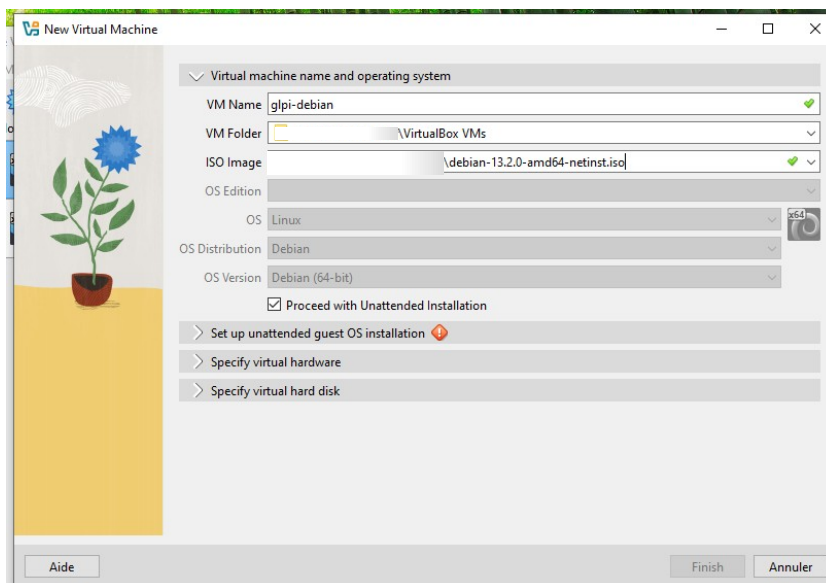
- Les étapes détaillées de installation système et applicative
- Les configurations réseau et DNS
- Les vérifications fonctionnelles
- Les bonnes pratiques de sécurisation et d'administration
- Les sources

Chaque étape est illustrée par une capture d'écran, des commandes exactes et des références techniques afin de garantir la reproductibilité et la compréhension du processus.

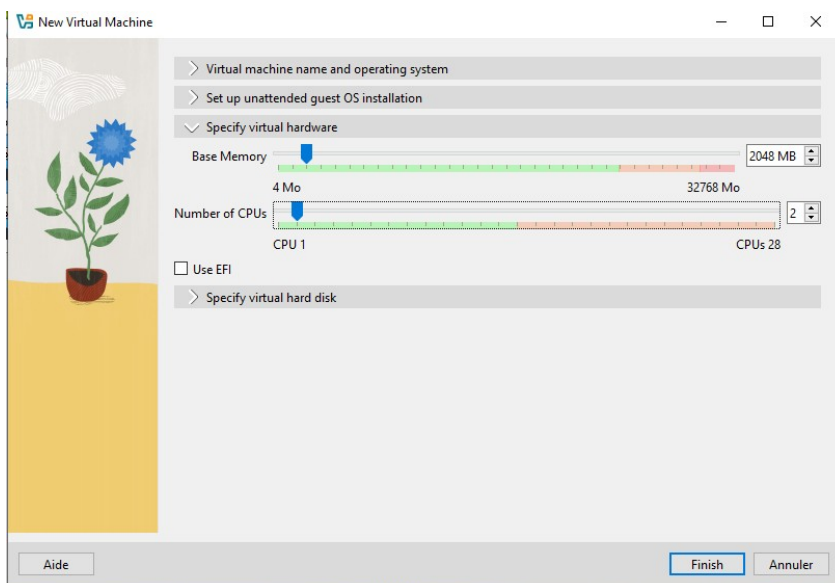
Étape 1 : Création de la machine virtuelle

Pour la création de la machine virtuelle, j'utilise **Oracle VirtualBox**. J'ai choisi de la nommer : **glpi-debian** pour l'image disque . Puis j'ai téléchargé sur le site officiel de debian : **debian-11.6.0-amd64-netinst.iso**. Pour le mode réseau, j'ai choisi un **Accès par pont** (l'interface est enp0s3), le stockage est sur **disque SATA 20Go plus l'ISO monté sur IDE**, la langue et le pays sont **Français et France**, le **partitionnement** est Assisté, tout dans une seule partition. Le mot de passe root est défini manuellement, pour l'utilitaire système, j'ai sélectionné les paquets de base uniquement.

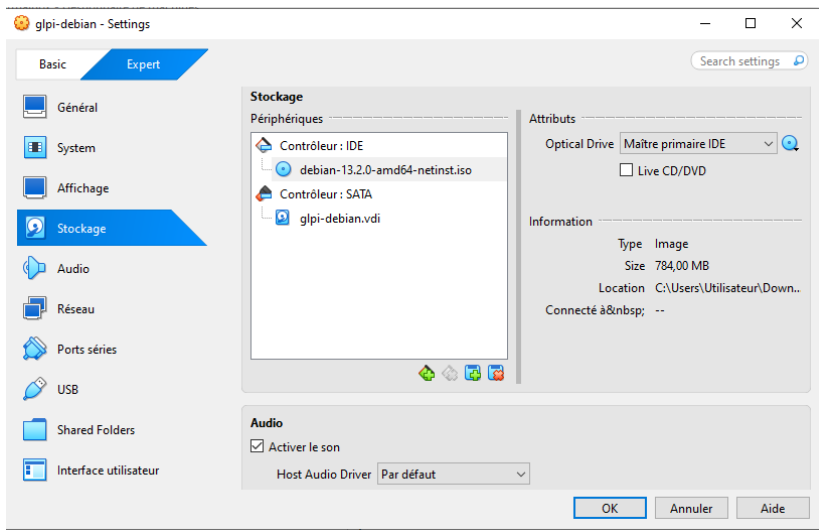
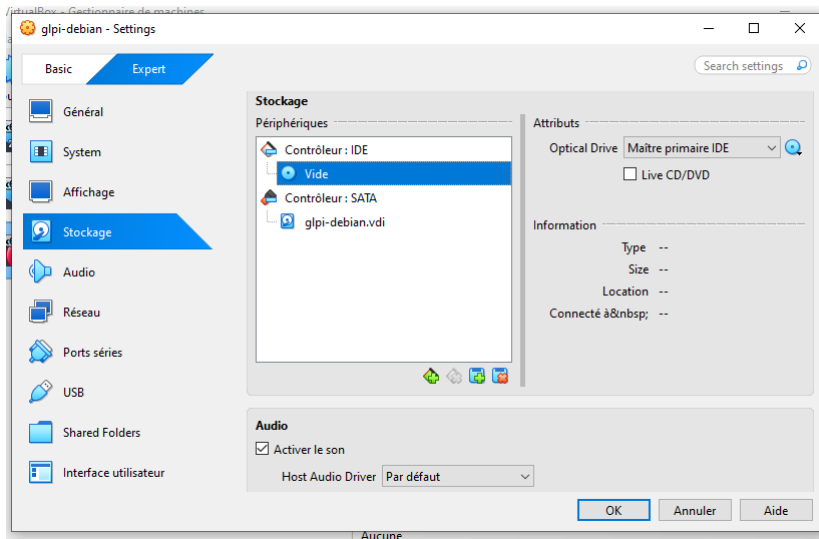
1. Configuration de base



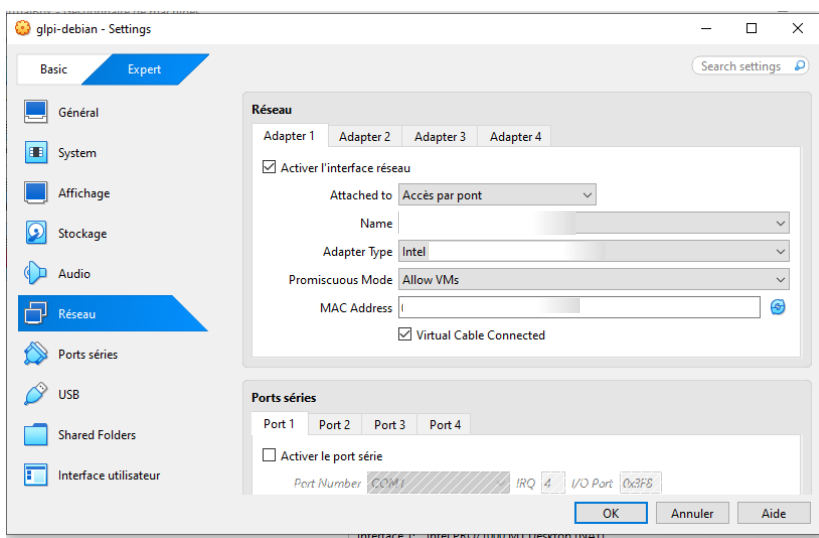
2. Allocation des ressources



3.Montage de l'ISO sur l'IDE

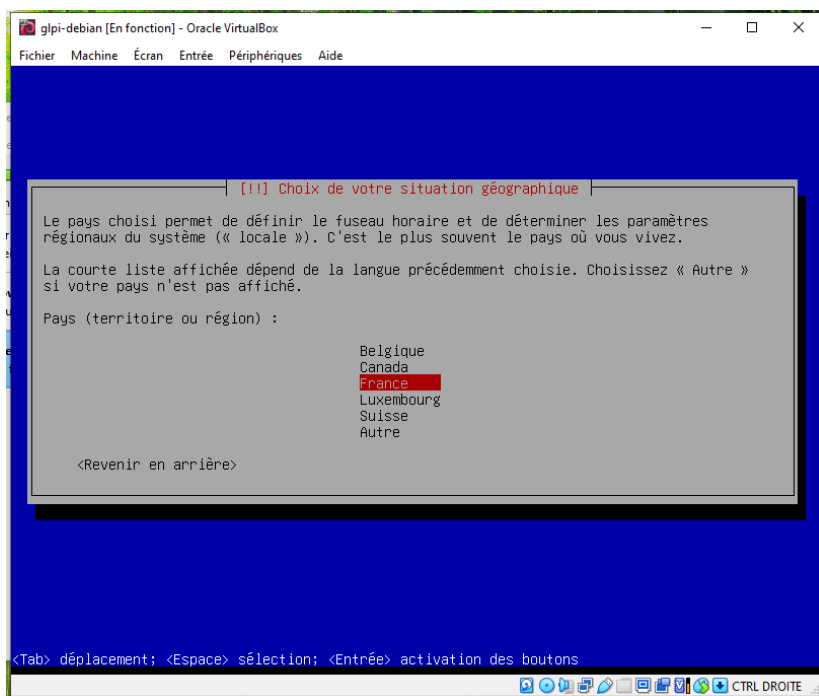
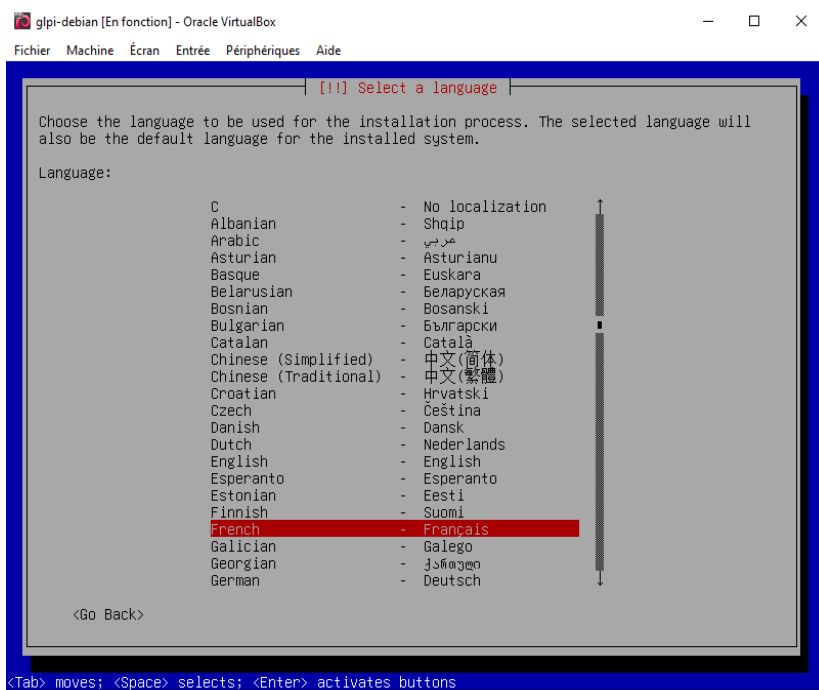


4.Choix du mode réseau

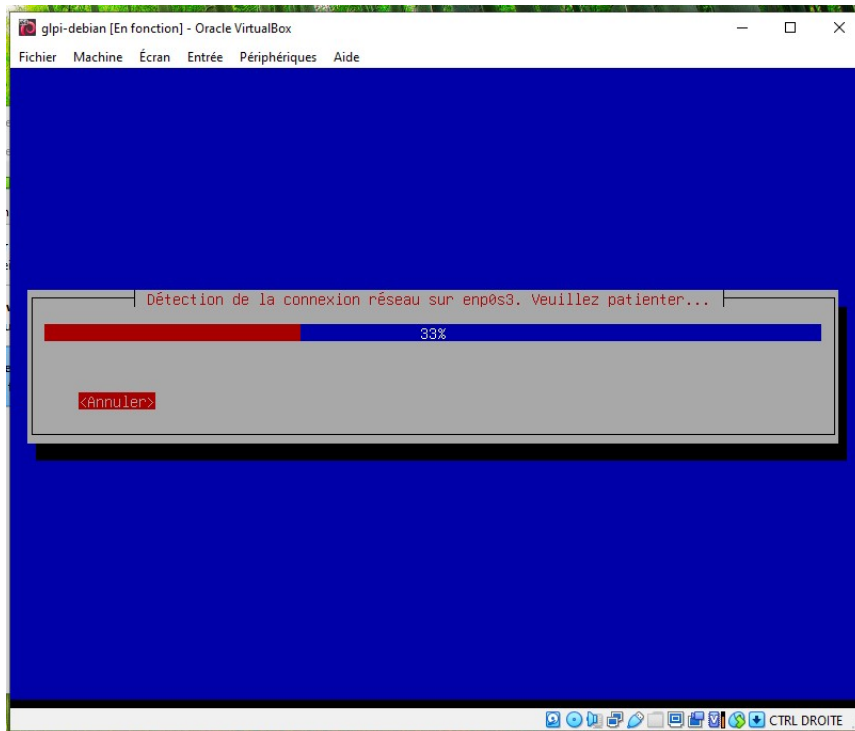


Installation non graphique de Debian :

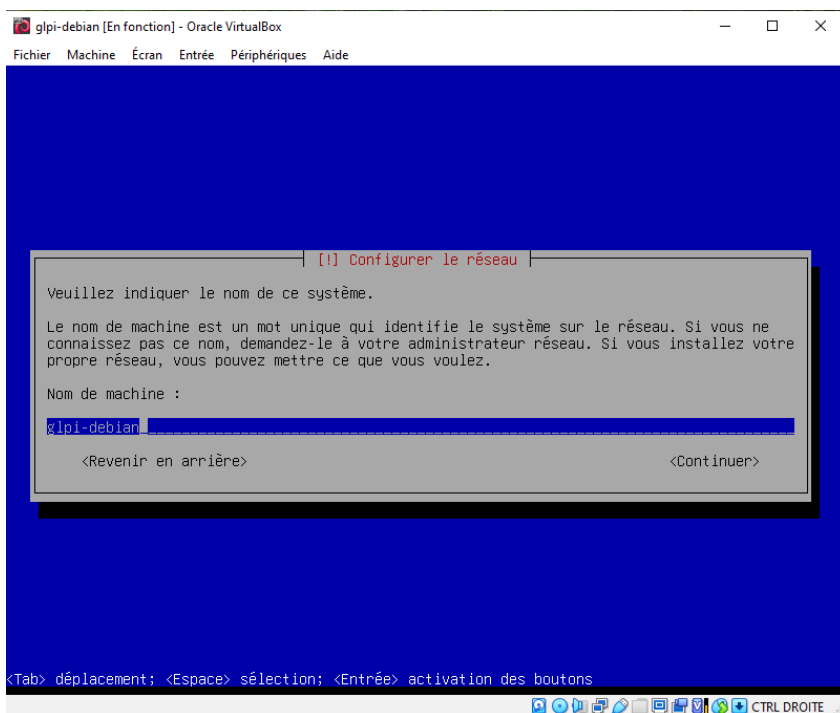
1.2 choix des langues



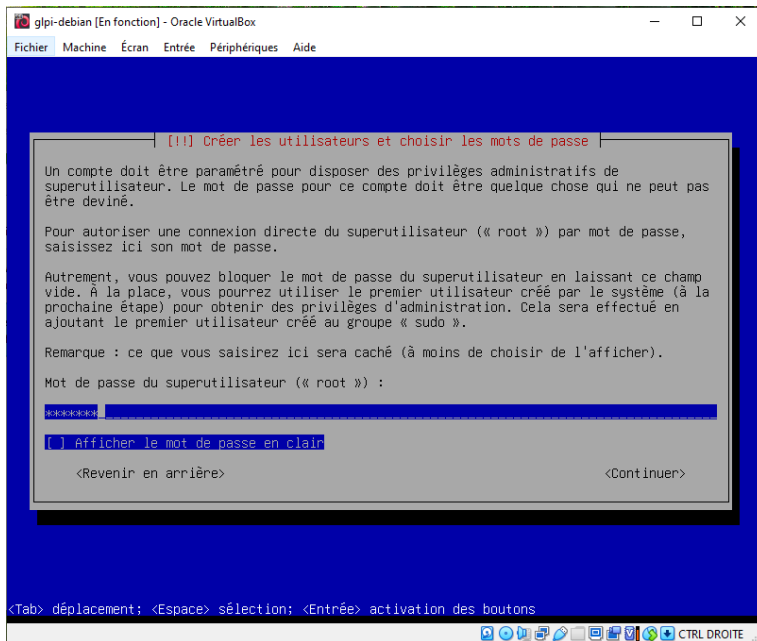
1.3 Gestion automatique de la connexion web :



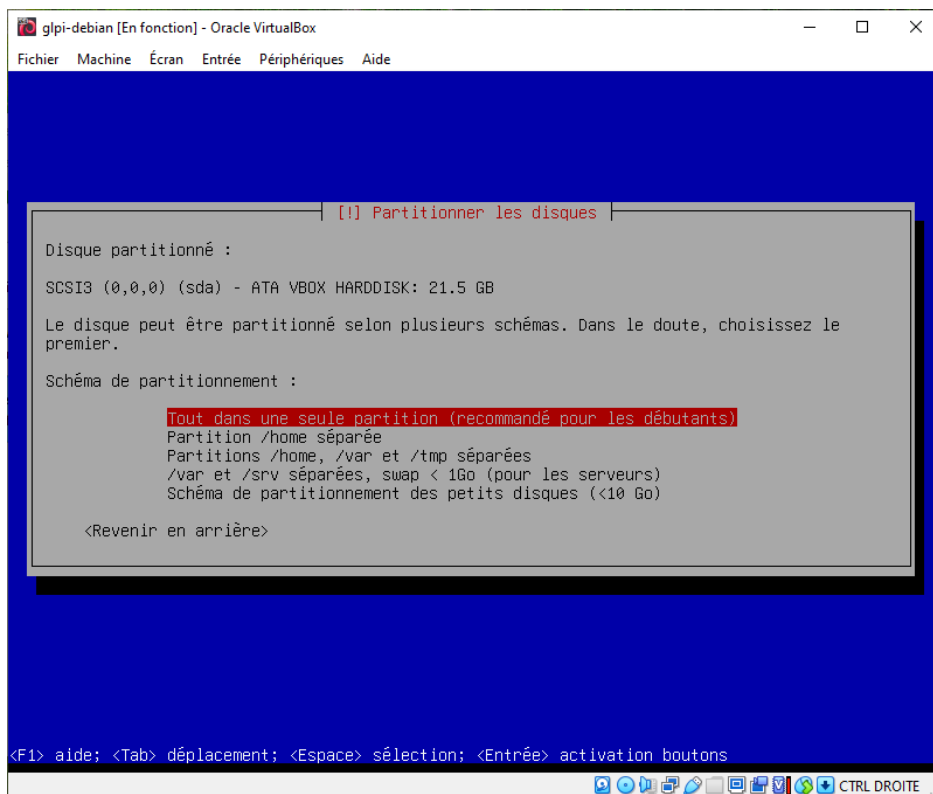
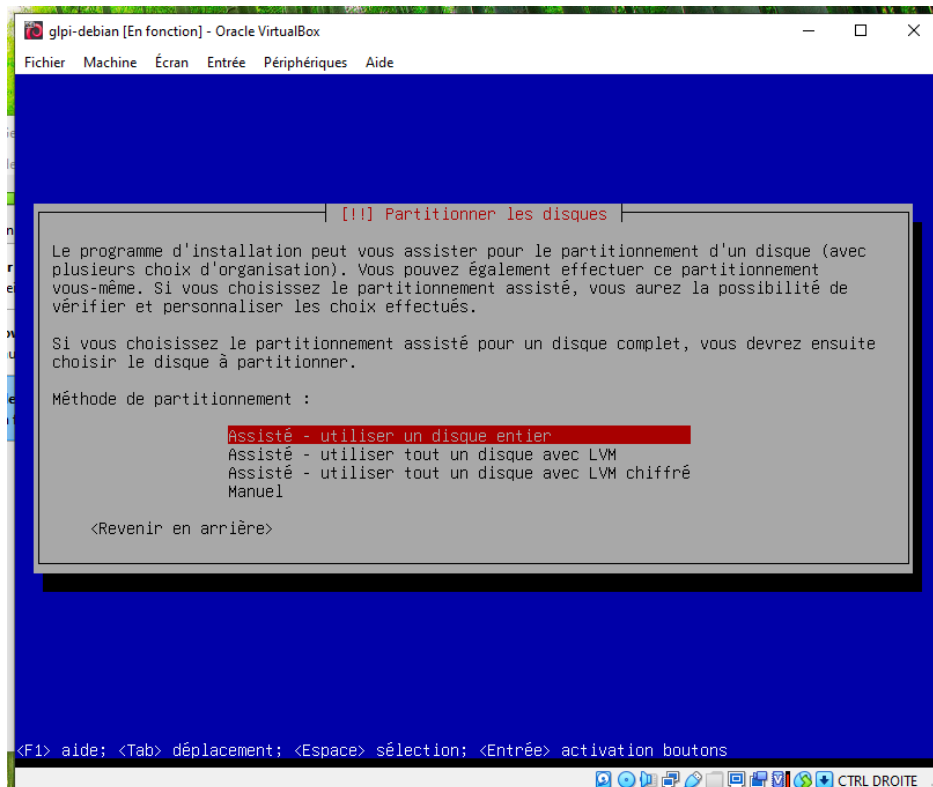
1.4 Nom de la machine

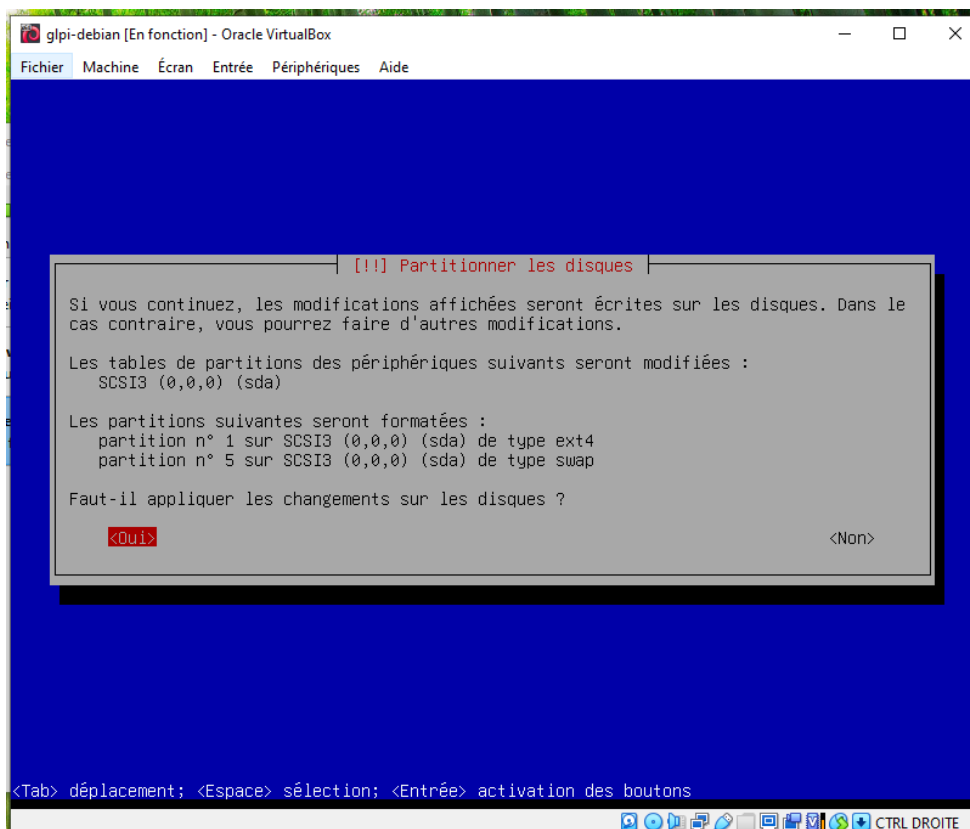
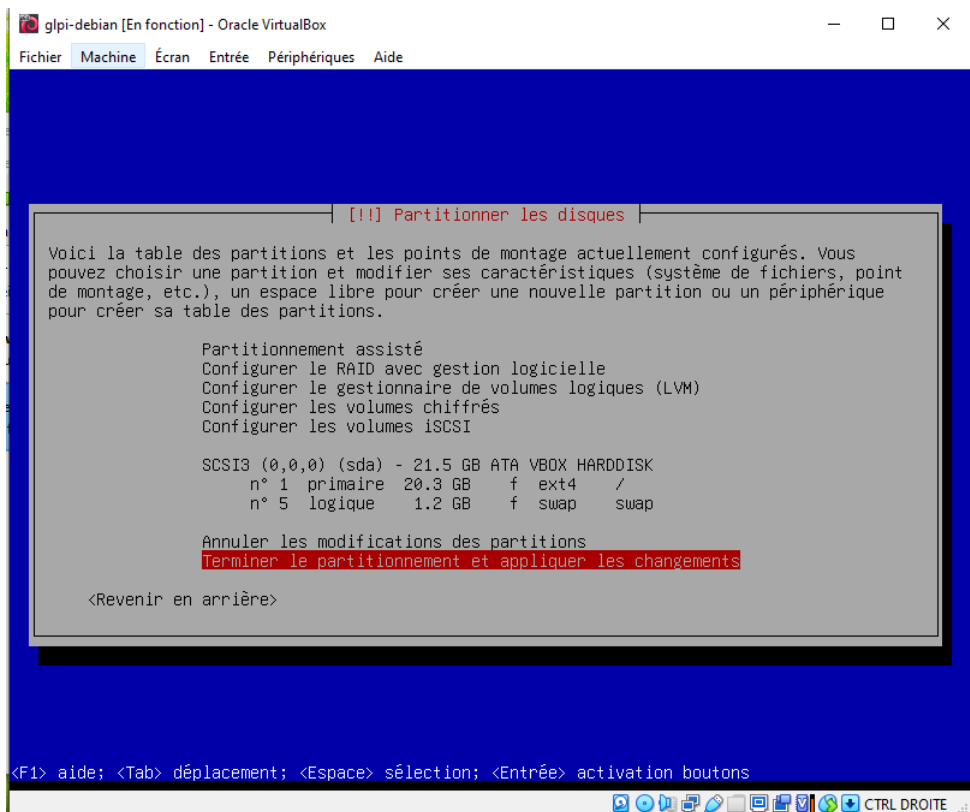


1.5 Configuration du ROOT

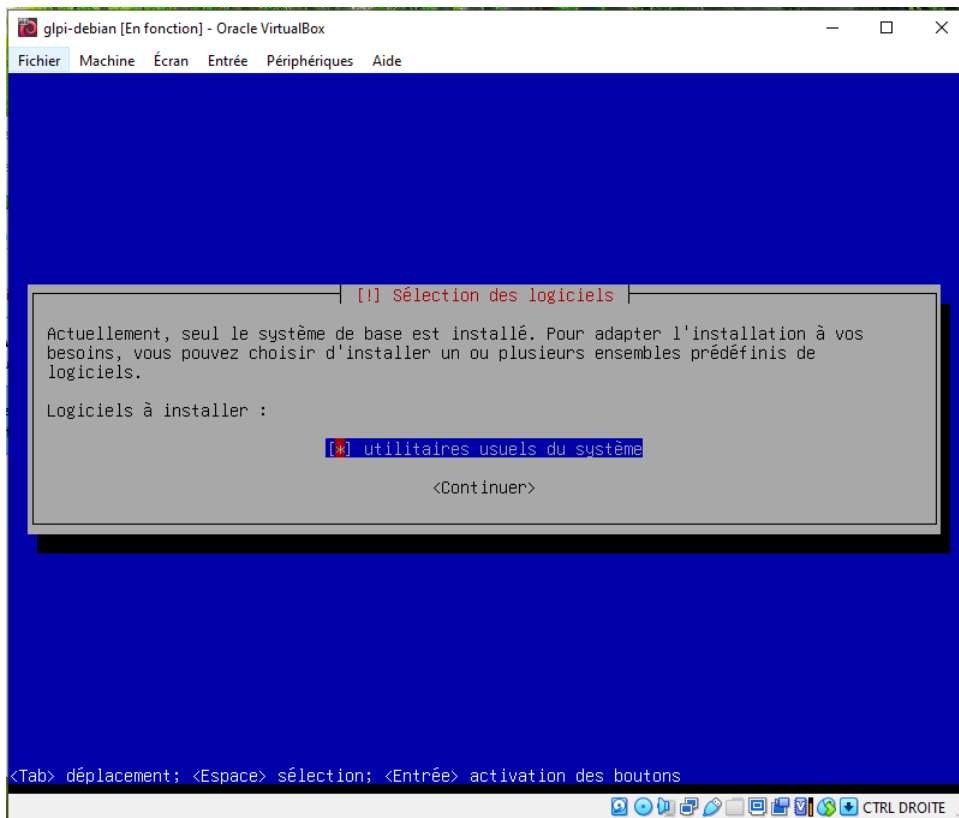


1.6 Gestion de disque/installation

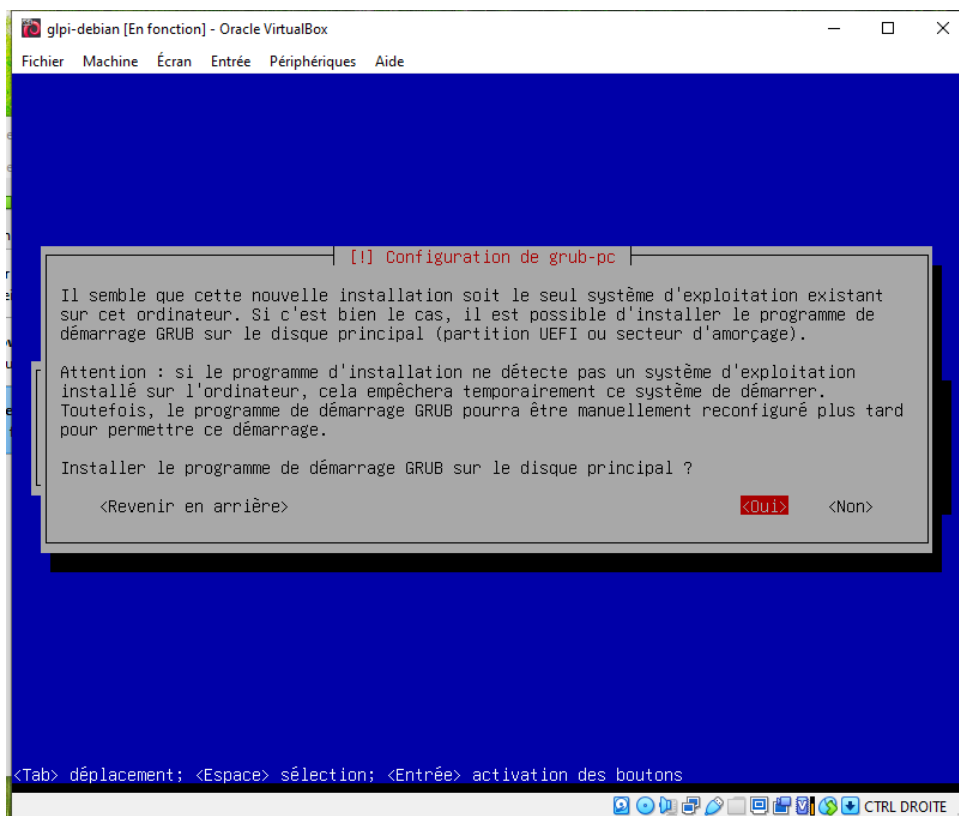




1.7 Installation des paquets



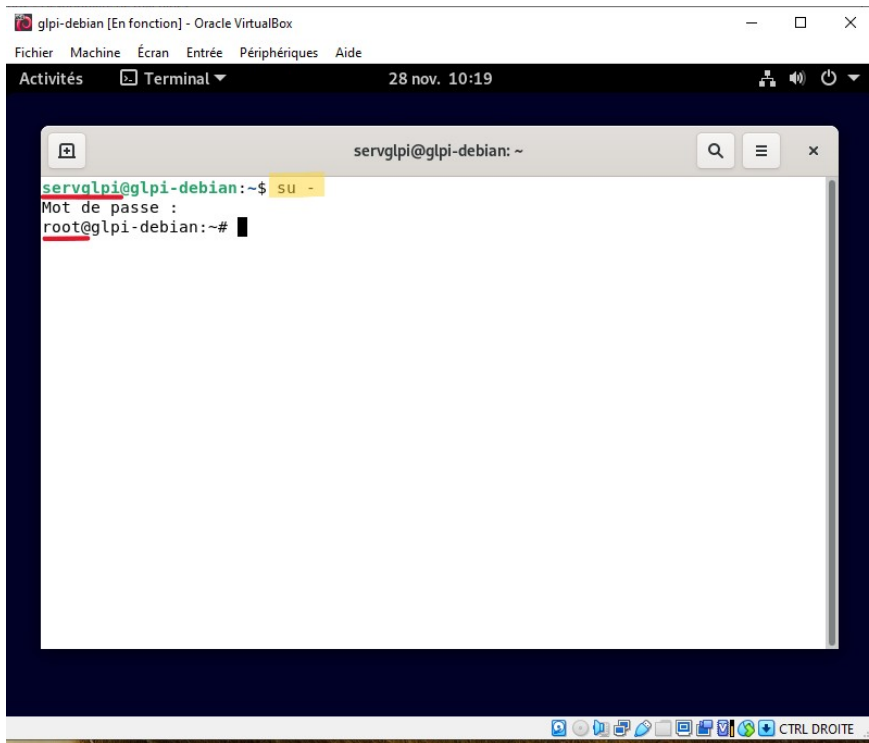
1.8 Installation de GRUB

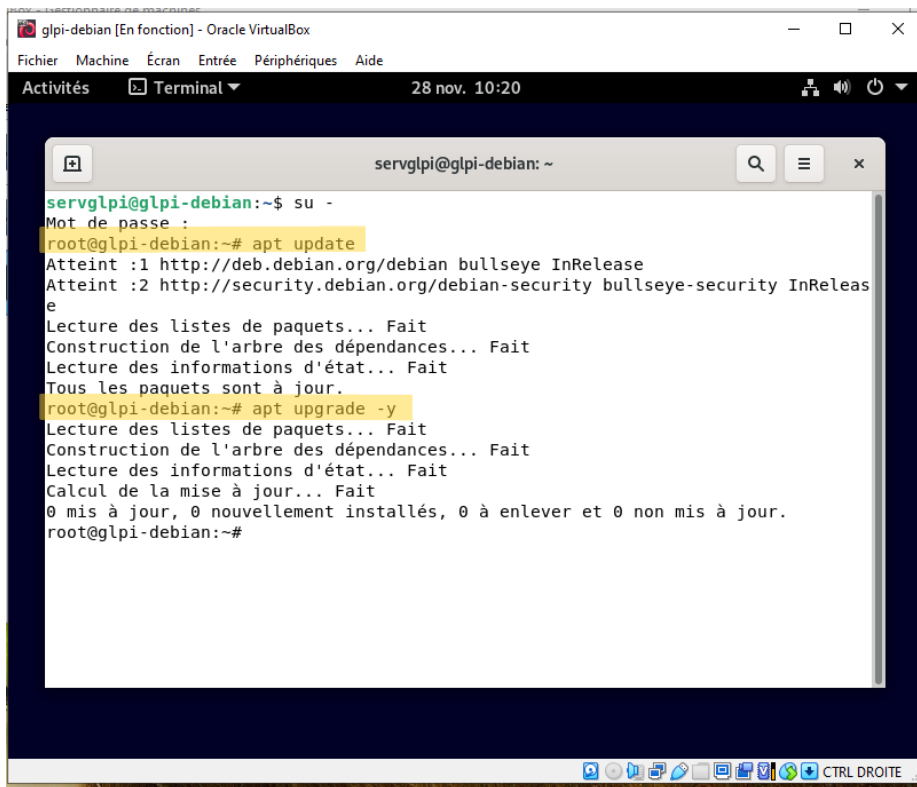


Étape 2 : Accès au ROOT et mise à jour du système

Dans cette étape, il est décrit de manière simple comment accéder au super user (ROOT) sur Debian. Mais également la mise à jour des paquets ainsi que leurs installations.

Le choix a été fait d'être en ROOT tout au long de cet exercice ce qui permet d'avoir toutes les autorisations sans devoir taper des commandes inutiles ou rentrer constamment le mot de passe.





```
servglpi@glpi-debian:~$ su -
Mot de passe :
root@glpi-debian:~# apt update
Atteint :1 http://deb.debian.org/debian bullseye InRelease
Atteint :2 http://security.debian.org/debian-security bullseye-security InRelease
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Tous les paquets sont à jour.
root@glpi-debian:~# apt upgrade -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@glpi-debian:~#
```

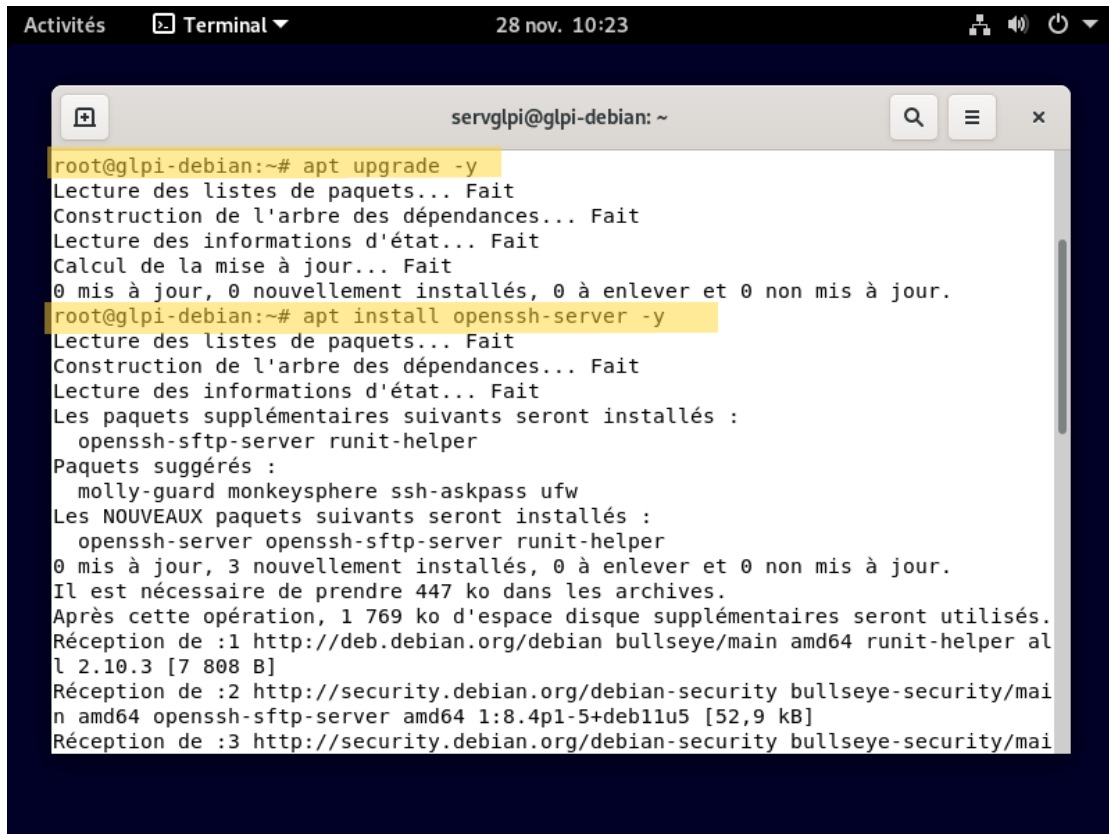
« **apt update** »: est une commande qui met à jour la liste de paquets disponibles (elle actualise la base de donnée locale des dépôts APT)

Elle va contacter toutes les sources définies dans le fichier `/etc/apt/sources.list` et `/etc/apt/sources.list.d/*` il est possible d'y accéder et de le modifier en ajoutant la commande « **nano** » devant les chemins.

« **apt upgrade -y** » : est une commande qui sert à installer les mises à jour disponibles pour les paquets déjà présents sur le système. Elle met à jour les paquets existants tout en préservant les fichiers de configuration gérant automatiquement les dépendances mais **attention** cette commande ne supprime jamais de paquet et n'installe pas de nouvelles dépendance majeures pour cela il faut taper la commande : « **apt full-upgrade** ».

L'option « -y » signifie « yes » automatique.

Étape 3 : Installation du serveur SSH

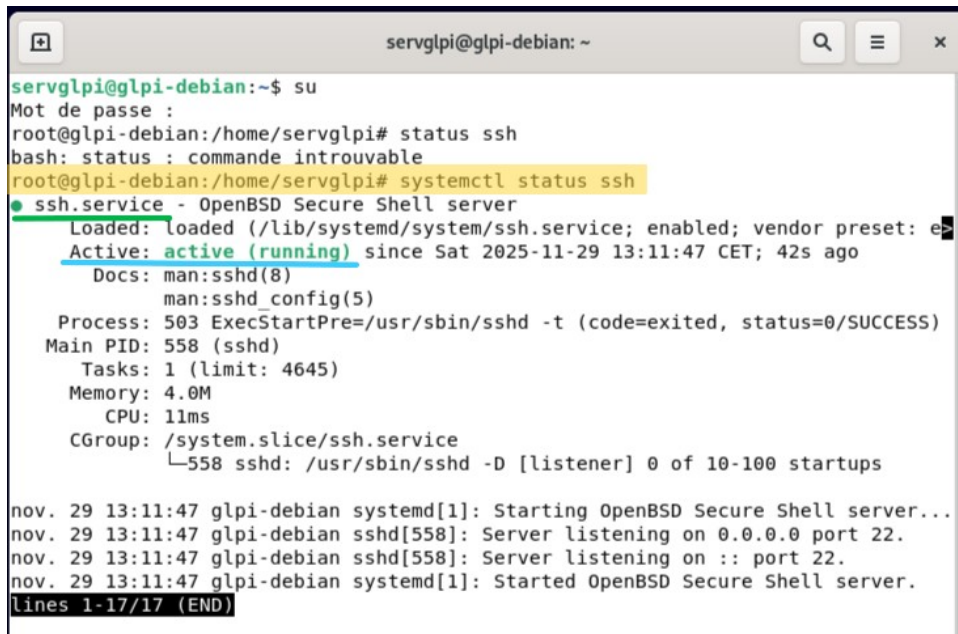


The screenshot shows a terminal window titled 'servglpi@glpi-debian: ~' with a dark background. The terminal output shows the execution of two commands: 'apt upgrade -y' and 'apt install openssh-server -y'. The first command updates the system packages. The second command installs the 'openssh-server' package along with its dependencies, including 'openssh-sftp-server' and 'runit-helper'. The terminal shows the progress of the installation, including the download of packages from the Debian repositories.

```
root@glpi-debian:~# apt upgrade -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Calcul de la mise à jour... Fait
0 mis à jour, 0 nouvellement installés, 0 à enlever et 0 non mis à jour.
root@glpi-debian:~# apt install openssh-server -y
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  openssh-sftp-server runit-helper
Paquets suggérés :
  molly-guard monkeysphere ssh-askpass ufw
Les NOUVEAUX paquets suivants seront installés :
  openssh-server openssh-sftp-server runit-helper
0 mis à jour, 3 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 447 ko dans les archives.
Après cette opération, 1 769 ko d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bullseye/main amd64 runit-helper al
l 2.10.3 [7 808 B]
Réception de :2 http://security.debian.org/debian-security bullseye-security/mai
n amd64 openssh-sftp-server amd64 1:8.4p1-5+deb11u5 [52,9 kB]
Réception de :3 http://security.debian.org/debian-security bullseye-security/mai
```

De la même manière que la commande « **apt upgrade -y** », la commande « **apt install** » installe le paquet spécifié ainsi que toutes les dépendances nécessaires. « **openssh-server** » est le nom du paquet (il fournit un service SSH fonctionnel, les utilitaires nécessaires au fonctionnement du serveur SSH). Les fichiers de la configuration sont dans : **/etc/ssh/sshd_config** , « **-y** » est un oui automatique.

Pour vérifier que le serveur fonctionne la commande est : « **systemctl status ssh** »



```
servglpi@glpi-debian: ~  
servglpi@glpi-debian:~$ su  
Mot de passe :  
root@glpi-debian:/home/servglpi# status ssh  
bash: status : commande introuvable  
root@glpi-debian:/home/servglpi# systemctl status ssh  
● ssh.service - OpenBSD Secure Shell server  
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: e  
   Active: active (running) since Sat 2025-11-29 13:11:47 CET; 42s ago  
     Docs: man:sshd(8)  
           man:sshd_config(5)  
  Process: 503 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)  
 Main PID: 558 (sshd)  
    Tasks: 1 (limit: 4645)  
  Memory: 4.0M  
     CPU: 11ms  
   CGroup: /system.slice/ssh.service  
           └─558 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups  
  
nov. 29 13:11:47 glpi-debian systemd[1]: Starting OpenBSD Secure Shell server...  
nov. 29 13:11:47 glpi-debian sshd[558]: Server listening on 0.0.0.0 port 22.  
nov. 29 13:11:47 glpi-debian sshd[558]: Server listening on :: port 22.  
nov. 29 13:11:47 glpi-debian systemd[1]: Started OpenBSD Secure Shell server.  
lines 1-17/17 (END)
```

Cette commande permet d’afficher l’état du service ssh sur une machine Debian ou tout autre services qui utilise systemd.

« **systemctl** » : Est l’outil de gestion des services et des unités systemd, il permet de démarrer et arrêter un service, le redémarrer vérifier son statut etc

« **status** » : Est l’option qui demande a systemctl d’afficher le statut détaillé d’un service

« **ssh** » : Est le nom du service SSH

Étape 4 : Configuration réseau statique

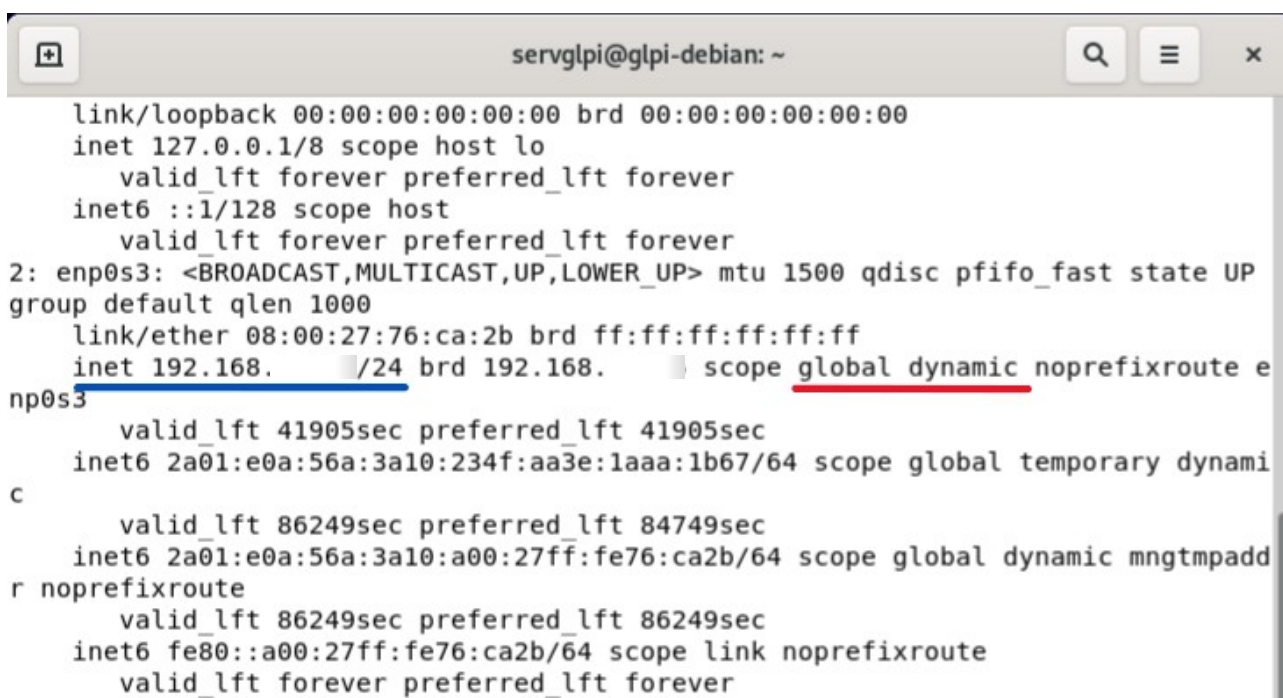
Dans cette étape, j'ai décidé de configurer un réseau avec une adresse IP statique car le serveur doit être joignable toujours à la même adresse, ce qui n'est pas le cas avec une IP dynamique car l'adresse peut changer à chaque démarrage, être réattribué après une expiration du bail ou être récupéré par une autre machine si le serveur n'est pas présent lors de l'attribution . C'est aussi indispensable pour un serveur qui héberge des services que la résolution DNS sois stable, avec des règles firewall précise et des routes réseau fixe. L'administration du réseau est aussi simplifiée pour la supervision des logs, la gestion des alertes, récupération en cas de crash etc. Cela évite aussi les conflits.

Attention ! Il faut choisir entre deux options : NetWorkManager ou /etc/network/interfaces car les deux cherchent à gérer l'interface.

Pour cet exercice, j'ai choisi d'utiliser NetWorkManager pour la configuration réseau du serveur, car il s'agit de la méthode moderne et recommandée dans les versions récentes de Debian. NetWorkManager offre une gestion dynamique, cohérente et centralisée des interfaces, une compatibilité étendue (VLAN, VPN, virtualisation) et un outil en ligne de commande puissant. À l'inverse, la configuration par /etc/network/interfaces est désormais considérée comme obsolète et peut entraîner des conflits avec des services de réseaux modernes.

Pour éviter tout conflits d'adressage IP, j'ai d'abord utilisé l'attribution d'adresse IP automatique pour avoir une adresse IP qui n'est utilisée par aucune autre machine.

L'adresse IP attribuée est 192.168.x.xxx /24 pour le voir, j'ai tapé la commande « **ip a** » ce qui donne le résultat suivant :



```
servglpi@glpi-debian: ~
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 ::1/128 scope host
    valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000
    link/ether 08:00:27:76:ca:2b brd ff:ff:ff:ff:ff:ff
    inet 192.168. /24 brd 192.168. scope global dynamic noprefixroute e
np0s3
    valid_lft 41905sec preferred_lft 41905sec
    inet6 2a01:e0a:56a:3a10:234f:aa3e:1aaa:1b67/64 scope global temporary dynami
c
    valid_lft 86249sec preferred_lft 84749sec
    inet6 2a01:e0a:56a:3a10:a00:27ff:fe76:ca2b/64 scope global dynamic mngtmpadd
r noprefixroute
    valid_lft 86249sec preferred_lft 86249sec
    inet6 fe80::a00:27ff:fe76:ca2b/64 scope link noprefixroute
    valid_lft forever preferred_lft forever
```

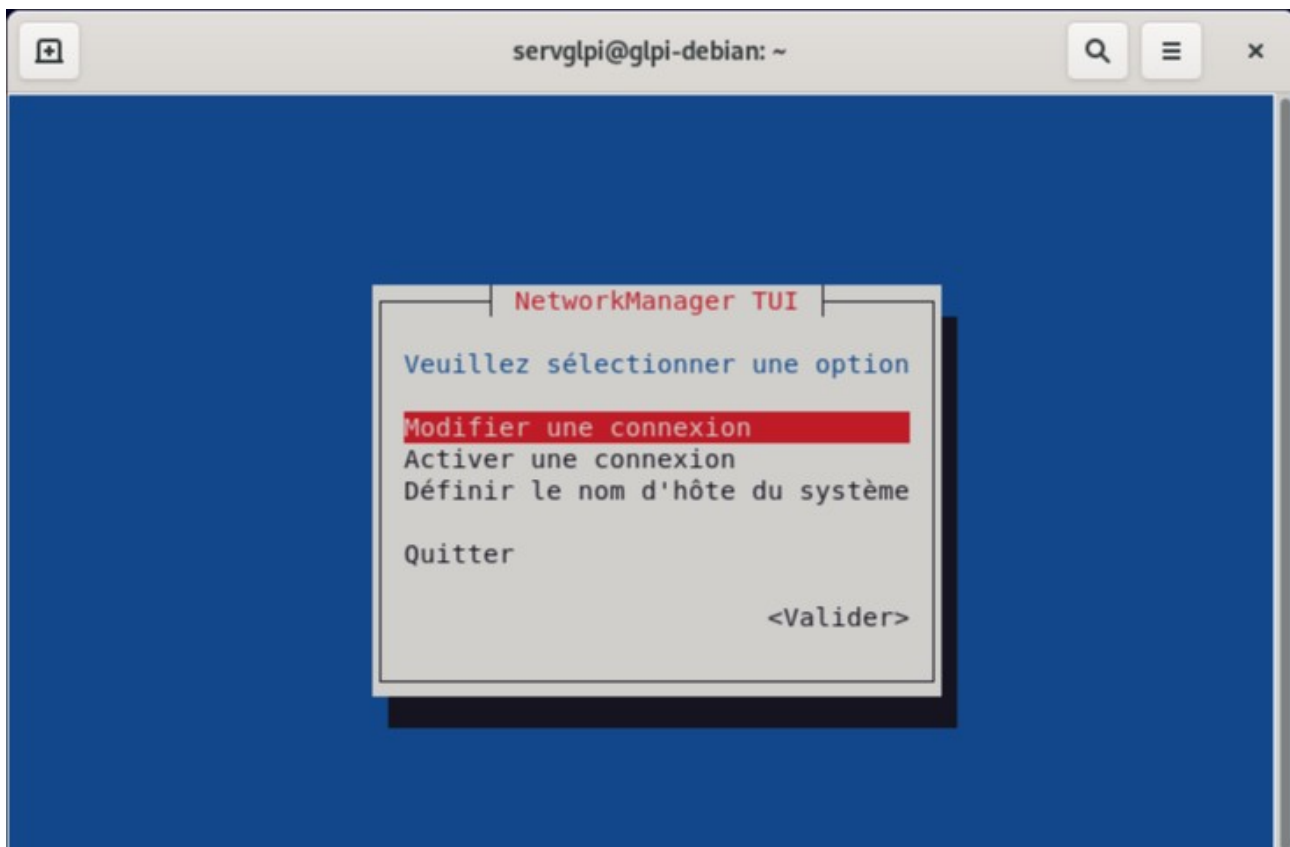
Ici l'adressage est bien dynamique.

Pour modifier cela une fois que j'obtiens l'adresse, je tape la commande « **nmtui** »

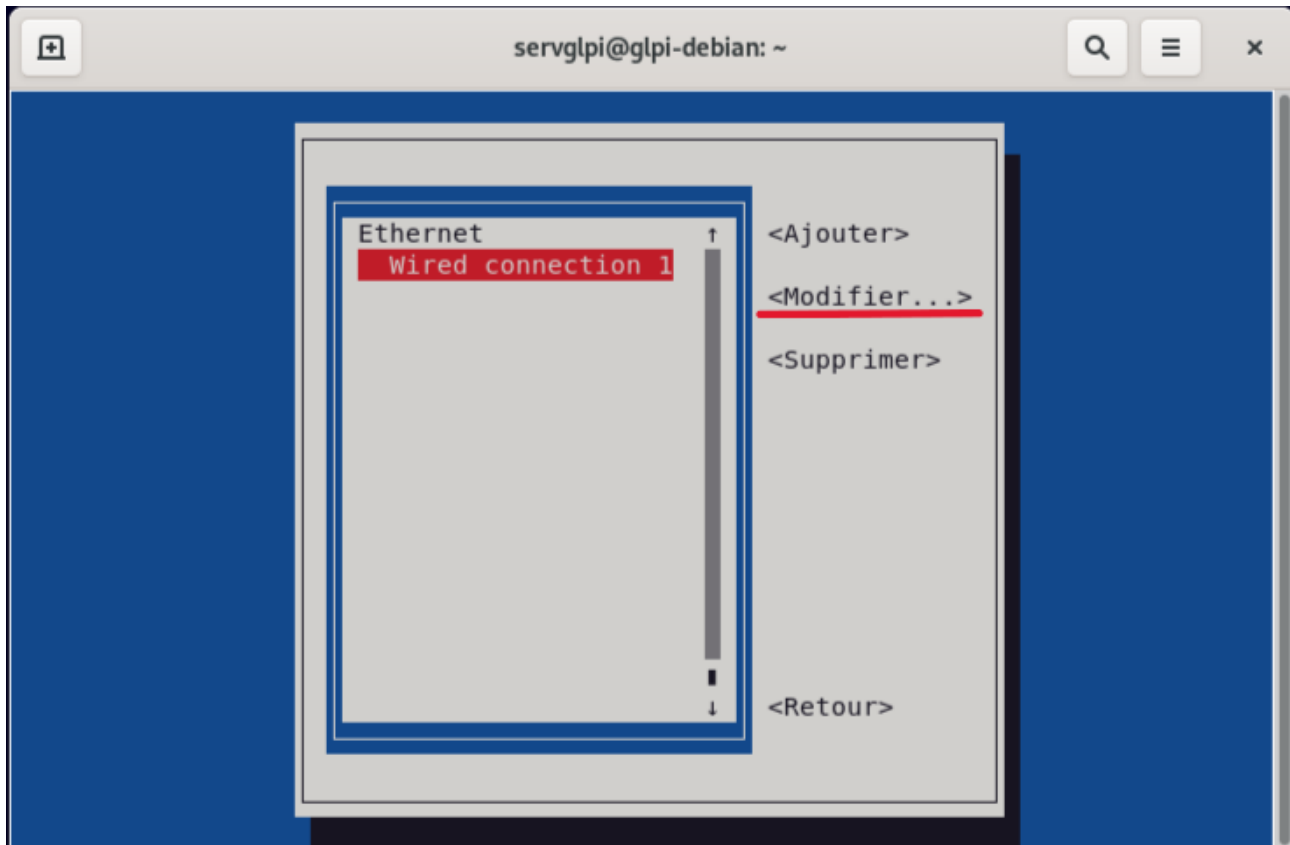
```
root@glpi-debian:/home/servglpi# nmtui
```

Cet outil a été utilisé pour configurer facilement et rapidement l'adresse IP statique du serveur. Il fournit une interface en mode texte intuitive permettant de modifier les paramètres réseau sans éditer manuellement des fichiers, réduisant ainsi les risques d'erreurs de syntaxe et offrant une gestion plus moderne et fiable du réseau.

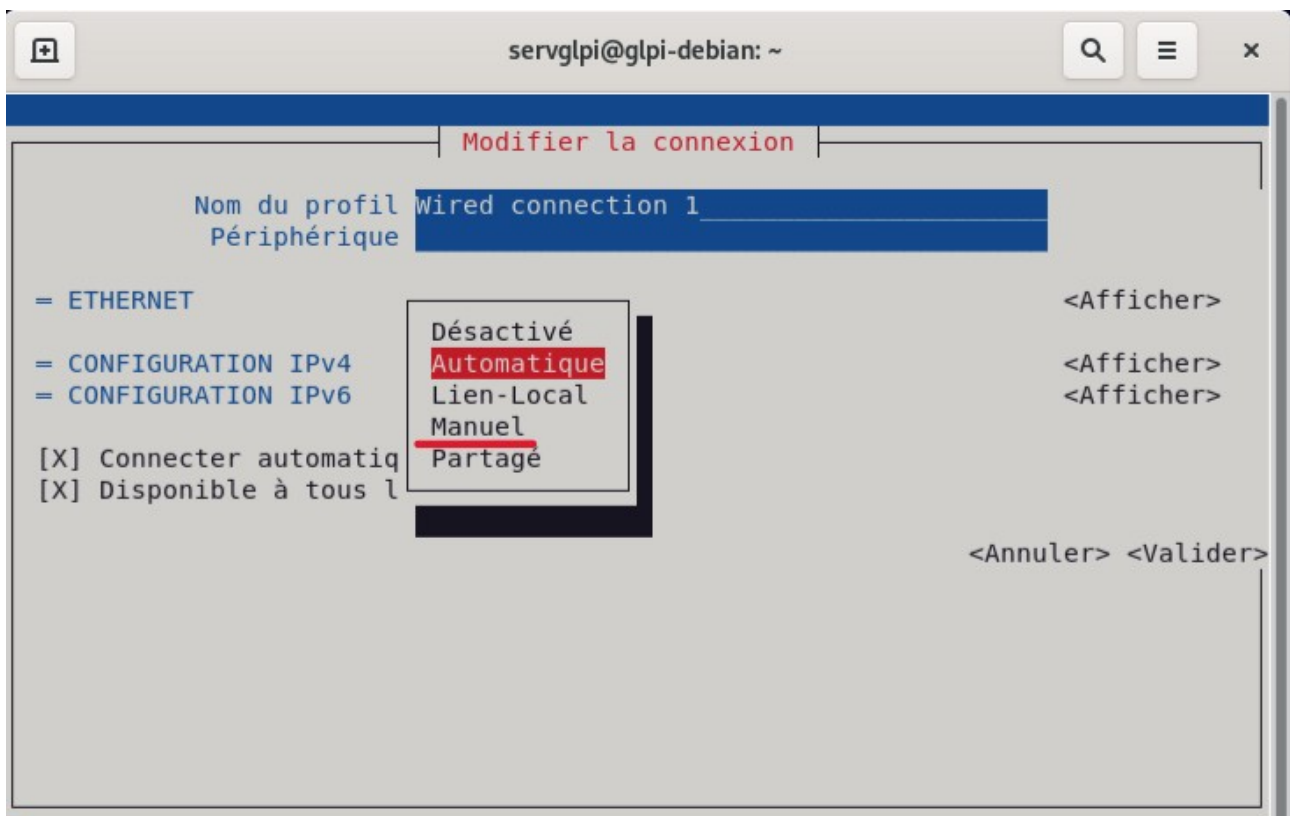
4.1 Modifier une connexion



4.2 Choisir la bonne connexion



4.3 Passer en manuel



4.4 Rentrer les informations obtenues et enregistrer.

servglpi@glpi-debian: ~

Modifier la connexion

Nom du profil: Wired connection 1

Périphérique: [selected]

= ETHERNET <Afficher>

CONFIGURATION IPv4: <Manuel> <Masquer>

Adresses: 192.168. /24 <Supprimer> <Ajouter...>

Passerelle: 192.168. <Supprimer> <Ajouter...>

Serveurs DNS: 8.8.8.8 <Supprimer> 8.8.4.4 <Supprimer> <Ajouter...>

Domaines de recherche: <Ajouter...>

Routage (aucune route personnalisée) <Modifier...>

☐ Ne jamais utiliser ce réseau comme route par défaut

☐ Ignorer les routes obtenues automatiquement

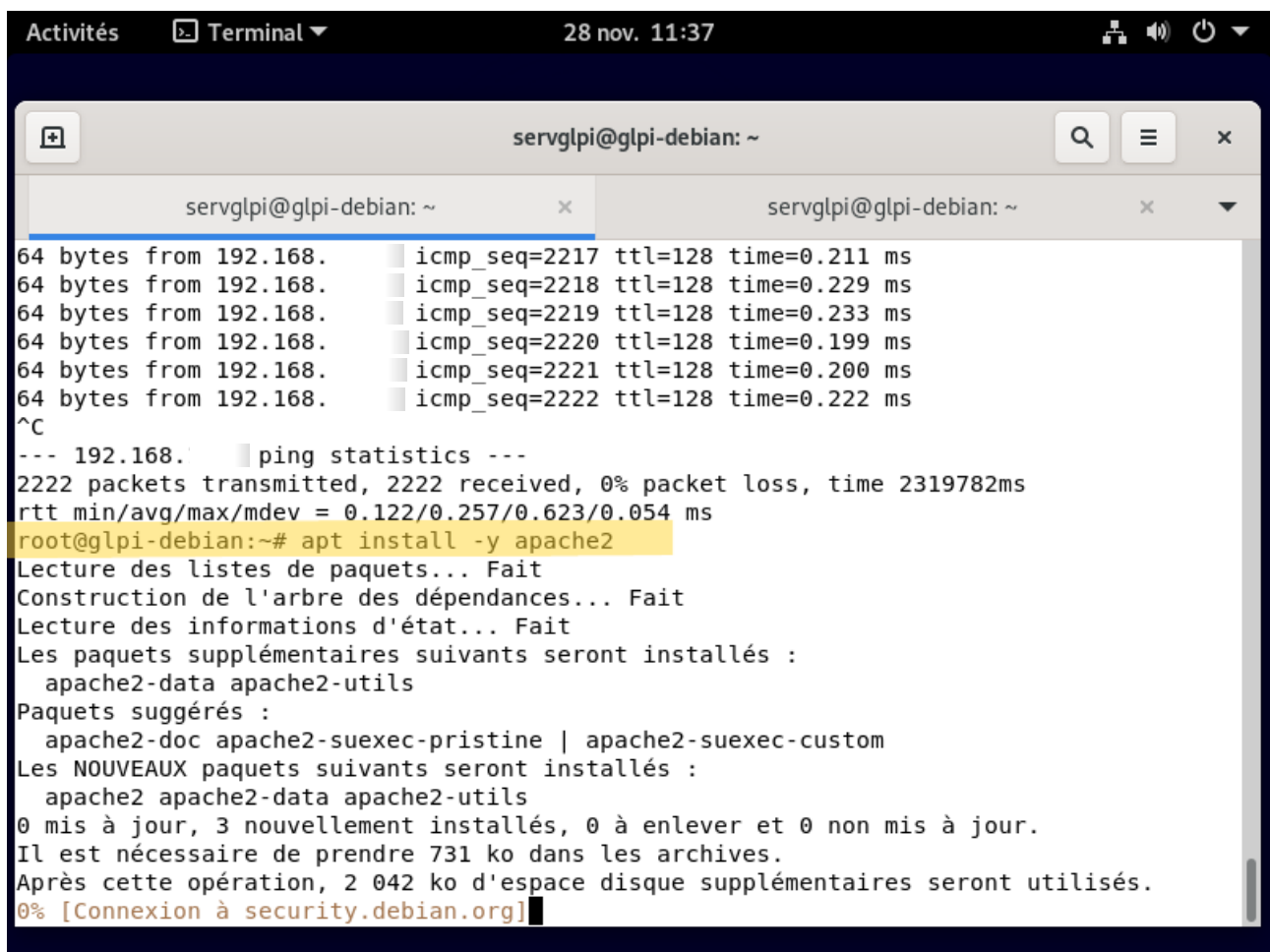
☐ Ignorer les paramètres DNS obtenus automatiquement

Étape 5 : Installation du serveur

Afin de préparer l'environnement nécessaire à l'installation et au fonctionnement de GLPI, il faut installer les composants essentiels de la pile LAMP.

- Apache2 et ses utilitaires pour assurer le rôle de serveur web
- MariaDB, une base de données performante, stable et compatible MySQL
- PHP ainsi que les modules indispensables (mysql, curl, intl, gd, xml, mbstring, ldap, apcu) pour garantir la compatibilité complète avec GLPI

5.1 Installation Apache2



```
Activités Terminal 28 nov. 11:37
servglpi@glpi-debian: ~
servglpi@glpi-debian: ~
64 bytes from 192.168.    icmp_seq=2217 ttl=128 time=0.211 ms
64 bytes from 192.168.    icmp_seq=2218 ttl=128 time=0.229 ms
64 bytes from 192.168.    icmp_seq=2219 ttl=128 time=0.233 ms
64 bytes from 192.168.    icmp_seq=2220 ttl=128 time=0.199 ms
64 bytes from 192.168.    icmp_seq=2221 ttl=128 time=0.200 ms
64 bytes from 192.168.    icmp_seq=2222 ttl=128 time=0.222 ms
^C
--- 192.168.    ping statistics ---
2222 packets transmitted, 2222 received, 0% packet loss, time 2319782ms
rtt min/avg/max/mdev = 0.122/0.257/0.623/0.054 ms
root@glpi-debian:~# apt install -y apache2
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  apache2-data apache2-utils
Paquets suggérés :
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
Les NOUVEAUX paquets suivants seront installés :
  apache2 apache2-data apache2-utils
0 mis à jour, 3 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 731 ko dans les archives.
Après cette opération, 2 042 ko d'espace disque supplémentaires seront utilisés.
0% [Connexion à security.debian.org]
```

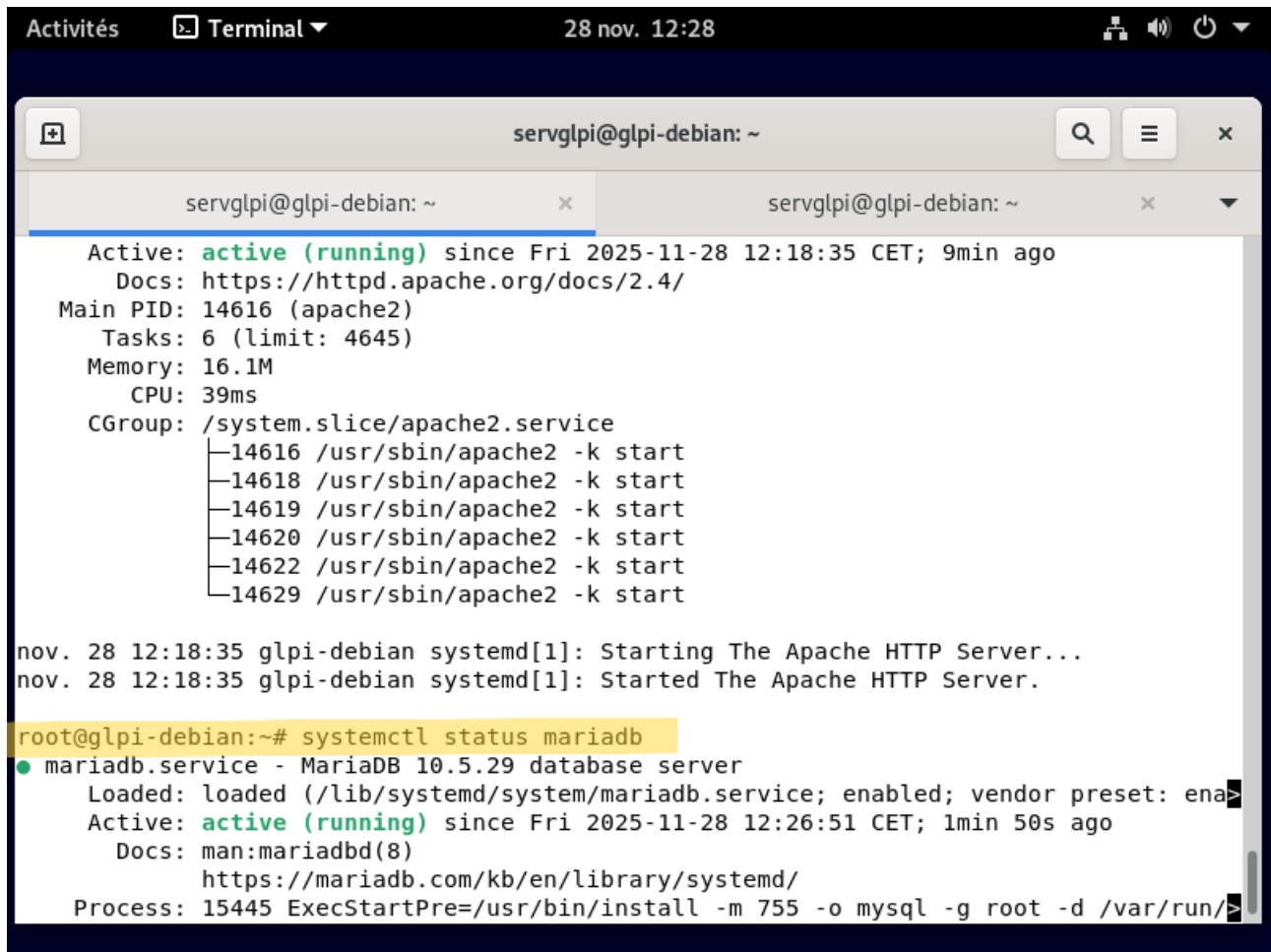
5.2 Installation PHP et ses modules

```
root@glpi-debian:~# apt install -y php php-cli php-mysql php-curl php-gd php-intl php-xml php-mbstring php-ldap php-apcu
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  libapache2-mod-php7.4 libonig5 php-apcu-bc php-common php7.4 php7.4-cli
  php7.4-common php7.4-curl php7.4-gd php7.4-intl php7.4-json php7.4-ldap
  php7.4-mbstring php7.4-mysql php7.4-opcache php7.4-readline php7.4-xml
Paquets suggérés :
  php-pear
Les NOUVEAUX paquets suivants seront installés :
  libapache2-mod-php7.4 libonig5 php php-apcu php-apcu-bc php-cli php-common
  php-curl php-gd php-intl php-ldap php-mbstring php-mysql php-xml php7.4 php7.4-cli
  php7.4-common php7.4-curl php7.4-gd php7.4-intl php7.4-json php7.4-ldap
  php7.4-mbstring php7.4-mysql php7.4-opcache php7.4-readline php7.4-xml
0 mis à jour, 27 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 5 252 ko dans les archives.
Après cette opération, 21,9 Mo d'espace disque supplémentaires seront utilisés.
Réception de :1 http://deb.debian.org/debian bullseye/main amd64 php-common all 2:76 [
15,6 kB]
Réception de :2 http://security.debian.org/debian-security bullseye-security/main amd6
4 php7.4-common amd64 7.4.33-1+deb11u9 [1 027 kB]
Réception de :3 http://deb.debian.org/debian bullseye/main amd64 php-apcu amd64 5.1.19
```

5.3 Installation de MariaDB

```
Traitement des actions différées (« triggers ») pour man-db (2.9.4-2) ...
Traitement des actions différées (« triggers ») pour libc-bin (2.31-13+deb11u13) ...
Traitement des actions différées (« triggers ») pour php7.4-cli (7.4.33-1+deb11u9) ...
Traitement des actions différées (« triggers ») pour libapache2-mod-php7.4 (7.4.33-1+deb11u9) ...
root@glpi-debian:~# apt install -y mariadb-server mariadb-client
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  galera-4 gawk libaio1 libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl
  libdbd-mariadb-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0ldbl
  libhtml-template-perl libmariadb3 libsigsegv2 libterm-readkey-perl
  mariadb-client-10.5 mariadb-client-core-10.5 mariadb-common mariadb-server-10.5
  mariadb-server-core-10.5 mysql-common rsync socat
Paquets suggérés :
  gawk-doc libmldbm-perl libnet-daemon-perl libsql-statement-perl
  libipc-sharedcache-perl mailx mariadb-test netcat-openbsd
Les NOUVEAUX paquets suivants seront installés :
  galera-4 gawk libaio1 libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl
  libdbd-mariadb-perl libdbi-perl libfcgi-bin libfcgi-perl libfcgi0ldbl
  libhtml-template-perl libmariadb3 libsigsegv2 libterm-readkey-perl mariadb-client
  mariadb-client-10.5 mariadb-client-core-10.5 mariadb-common mariadb-server
  mariadb-server-10.5 mariadb-server-core-10.5 mysql-common rsync socat
```

5.4 Vérification du statut de MariaDB



```
Activités Terminal 28 nov. 12:28

servglpi@glpi-debian: ~
Active: active (running) since Fri 2025-11-28 12:18:35 CET; 9min ago
Docs: https://httpd.apache.org/docs/2.4/
Main PID: 14616 (apache2)
Tasks: 6 (limit: 4645)
Memory: 16.1M
CPU: 39ms
CGroup: /system.slice/apache2.service
├─14616 /usr/sbin/apache2 -k start
├─14618 /usr/sbin/apache2 -k start
├─14619 /usr/sbin/apache2 -k start
├─14620 /usr/sbin/apache2 -k start
├─14622 /usr/sbin/apache2 -k start
└─14629 /usr/sbin/apache2 -k start

nov. 28 12:18:35 glpi-debian systemd[1]: Starting The Apache HTTP Server...
nov. 28 12:18:35 glpi-debian systemd[1]: Started The Apache HTTP Server.

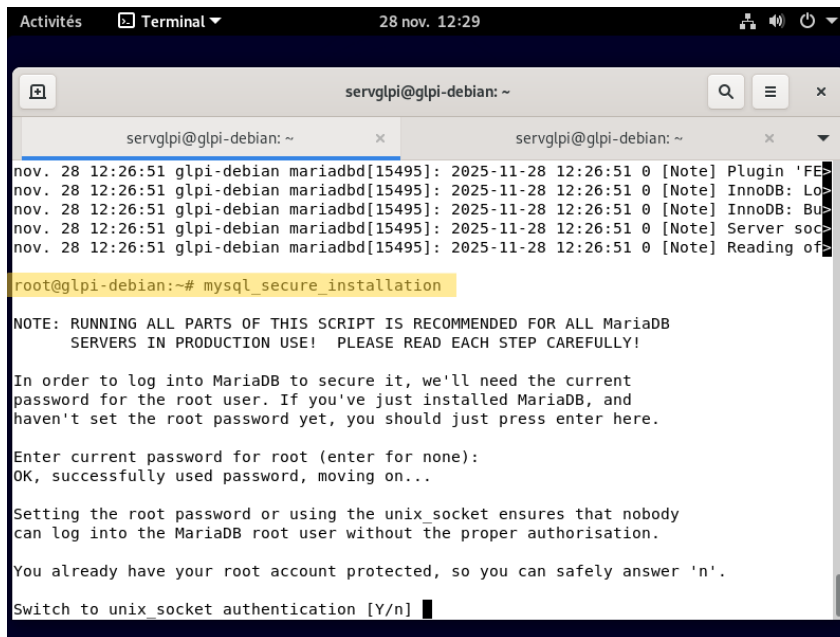
root@glpi-debian:~# systemctl status mariadb
● mariadb.service - MariaDB 10.5.29 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: ena>
   Active: active (running) since Fri 2025-11-28 12:26:51 CET; 1min 50s ago
     Docs: man:mariadb(8)
           https://mariadb.com/kb/en/library/systemd/
   Process: 15445 ExecStartPre=/usr/bin/install -m 755 -o mysql -g root -d /var/run/>
```

Pourquoi ce choix ?

Dans le cadre de l'installation de GLPI, il est nécessaire de disposer d'un environnement serveur fiable et compatible avec les exigences techniques de l'application. GLPI repose sur le slack LAMP (Linux, Apache, MariaDB, PHP) une architecture largement adoptée, éprouvée et adaptée aux environnements de production.

Étape 6 : Sécurisation de MariaDB

MariaDB est le moteur de base de données qui sera utilisé pour stocker l'ensemble des informations de GLPI (utilisateurs, inventaire, tickets, configurations...). Il s'agit d'un système fiable, performant et totalement open-source, reconnu pour sa stabilité dans les environnements professionnels. Cependant, par défaut MariaDB n'est pas sécurisée, Debian propose un outil intégré pour renforcer immédiatement la sécurité du service. Cette étape est indispensable avant toute mise en production.



```
nov. 28 12:26:51 glpi-debian mariadb[15495]: 2025-11-28 12:26:51 0 [Note] Plugin 'FE>
nov. 28 12:26:51 glpi-debian mariadb[15495]: 2025-11-28 12:26:51 0 [Note] InnoDB: Lo>
nov. 28 12:26:51 glpi-debian mariadb[15495]: 2025-11-28 12:26:51 0 [Note] InnoDB: Bu>
nov. 28 12:26:51 glpi-debian mariadb[15495]: 2025-11-28 12:26:51 0 [Note] Server soc>
nov. 28 12:26:51 glpi-debian mariadb[15495]: 2025-11-28 12:26:51 0 [Note] Reading of>

root@glpi-debian:~# 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.

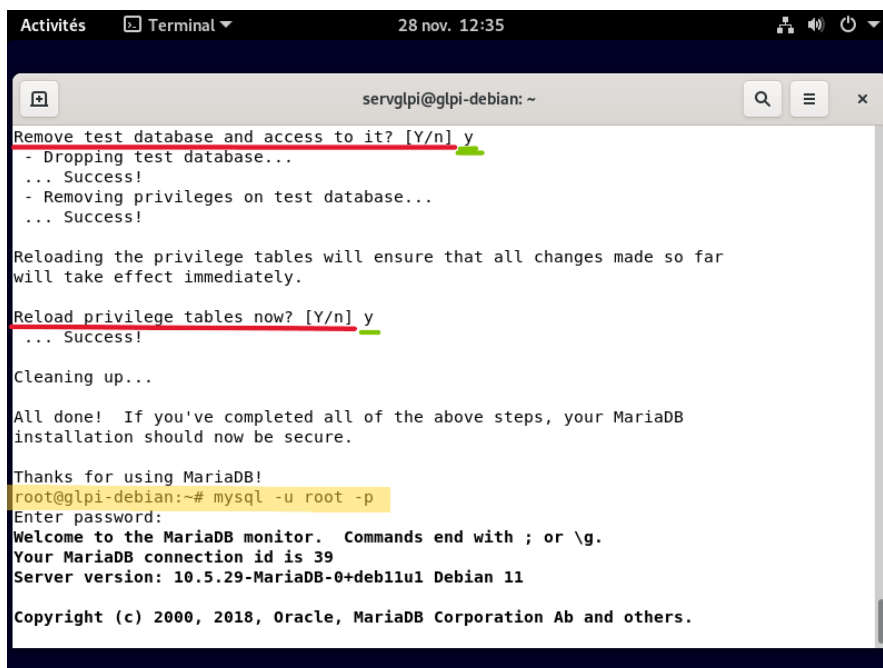
Enter current password for root (enter for none):
OK, successfully used password, moving on...

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]
```

Il faut ensuite nettoyer la base de donnée test et réinitialiser la table des mots de passes etc :



```
Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
root@glpi-debian:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.5.29-MariaDB-0+deb11u1 Debian 11

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

Étape 7 : Création de la base GLPI

Après l'installation et la sécurisation de MariaDB, il faut créer une base de données dédiée à GLPI ainsi qu'un utilisateur spécifique qui pourra y accéder. Cette approche respecte les bonnes pratiques de sécurité et de maintenance : chaque application possède son propre utilisateur et sa propre base.

- **Séparation des droits** : l'utilisateur SQL dédié à GLPI n'a accès qu'à sa base, ce qui limite l'impact en cas de compromission.
- **Encodage Unicode complet** : la base utilise **utf8mb4**, ce qui garantit le support des caractères spéciaux et émojis, indispensable pour l'inventaire multilingue ou les tickets utilisateur.
- **Reproductivité** : les commandes SQL peuvent être exécutées sur n'importe quel serveur Debian avec MariaDB, ce qui facilite la maintenance ou le déploiement automatique.

7.1 Connexion en ROOT

```
root@glpi-debian:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 39
Server version: 10.5.29-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.
```

Avant de créer la base de données GLPI et l'utilisateur, il faut se connecter à MariaDB en tant qu'administrateur.

« **mysql** » sert à interagir avec le serveur SQL, « **-u root** » est l'utilisateur root de MariaDB, « **-p** » demande le mot de passe.

7.2 Suppression de la base existante pour éviter les conflits

Pour cela on utilise la commande « **DROP DATABASE IF EXISTS glpidb ;** »

```
servglpi@glpi-debian: ~
MariaDB [(none)]> DROP DATABASE IF EXISTS glpidb;
Query OK, 0 rows affected, 1 warning (0,000 sec)
```


7.3 Suppression des utilisateurs

On supprime les utilisateurs existants pour repartir sur une base propre avec la commande « **DROP USER IF EXISTS 'glpiuser'@'localhost' ;** »

```
MariaDB [(none)]> DROP USER IF EXISTS 'glpiuser'@'localhost';  
Query OK, 0 rows affected, 1 warning (0,001 sec)
```

7.4 Création de la base

Avec la commande « **CREATE DATABASE glpidb CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci ;** » la base GLPI avec un encodage complet Unicode pour assurer la compatibilité multilingue est créée.

```
MariaDB [(none)]> CREATE DATABASE glpidb CHARACTER SET utf8mb4 COLLATE utf8mb4_u  
nicode_ci;  
Query OK, 1 row affected (0,001 sec)
```

7.5 Création d'un utilisateur dédié

Pour la création d'un utilisateur dédié GLPI et un mot de passe sécurisé il faut taper la commande « **CREATE USER 'glpiuser'@'localhost' IDENTIFIED BY 'motdepasse' ;** »

```
MariaDB [(none)]> CREATE USER 'glpiuser'@'localhost' IDENTIFIED BY 'comodor';  
Query OK, 0 rows affected (0,001 sec)
```

7.6 Privilèges

Pour donner tous les droits sur sa base tout en isolant le reste du serveur SQL , je tape la commande « **GRANT ALL PRIVILEGES ON glpidb.* TO 'glpiuser'@'localhost' ;** »

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON glpidb.* TO 'glpiuser'@'localhost';  
Query OK, 0 rows affected (0,001 sec)
```

7.7 Application et sortie de MariaDB

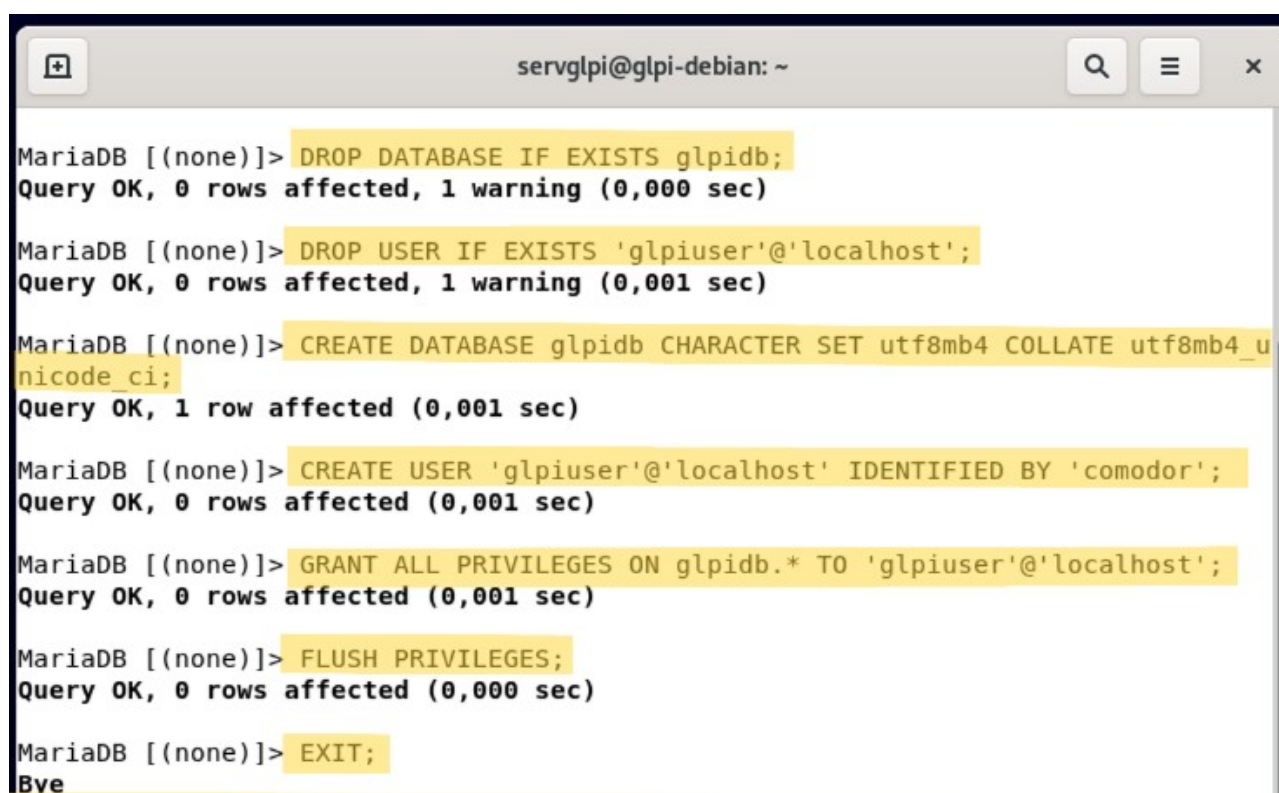
Il faut maintenant appliquer les droits et les changements effectués plus haut et pour ça j'utilise « **FLUSH PRIVILEGES ;** »

```
MariaDB [(none)]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0,000 sec)
```

Une fois cela fait, je peux sortir de la configuration grâce à « **EXIT ;** ».

```
MariaDB [(none)]> EXIT;  
Bye
```

La vision globale du terminal pendant la configuration :

A screenshot of a terminal window titled 'servglpi@glpi-debian: ~'. The terminal shows a series of MariaDB commands and their outputs. The commands are: 'DROP DATABASE IF EXISTS glpidb;', 'DROP USER IF EXISTS 'glpiuser'@'localhost';', 'CREATE DATABASE glpidb CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;', 'CREATE USER 'glpiuser'@'localhost' IDENTIFIED BY 'comodor';', 'GRANT ALL PRIVILEGES ON glpidb.* TO 'glpiuser'@'localhost';', 'FLUSH PRIVILEGES;', and 'EXIT;'. The outputs show the status of each query, such as 'Query OK, 0 rows affected' or 'Query OK, 1 row affected'. The terminal ends with 'Bye'.

Et pour terminer je teste la connexion.

```
root@glpi-debian:/home/servglpi# mysql -u glpiuser -p glpidb  
Enter password:  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 31  
Server version: 10.5.29-MariaDB-0+deb11u1 Debian 11
```


8.2 Décompresser

Maintenant je décompresse le fichier avec « **tar** ».

« **tar -xvf glpi-10.0.15.tgz** ».

« **tar** » est l'outil de gestion des archives, « **-x** » extrait les fichiers de l'archive, « **-v** » affiche les fichiers extraits, « **-f** » indique le nom du fichier à traiter.

8.3 Rendre le fichier lisible

Pour que GLPI soit accessible via le serveur Apache, on déplace le dossier dans « **/var/www/html/** »

```
root@glpi-debian:/tmp# tar -xzf glpi-10.0.15.tgz
root@glpi-debian:/tmp# glpi /var/www/html/
bash: glpi : commande introuvable
root@glpi-debian:/tmp# mv glpi /var/www/html/
```

« **mv** » déplace le dossier **glpi** vers le répertoire web par défaut d'apache.

Dans l'image ci-dessus j'ai fait une erreur de syntaxe j'ai oublié la commande « **mv** ».

8.4 Permissions et autorisations

Apache se sert de l'utilisateur et du groupe **www-data** pour accéder aux fichiers web. Il faut donc entrer les bonnes permissions :

```
root@glpi-debian:/tmp# chown -R www-data:www-data /var/www/html/glpi
root@glpi-debian:/tmp# chmod -R 755 /var/www/html/glpi
chmod: impossible d'accéder à '/var/www/html/glpi': Aucun fichier ou dossier de ce type
root@glpi-debian:/tmp# chmod -R 755 /var/www/html/glpi
```

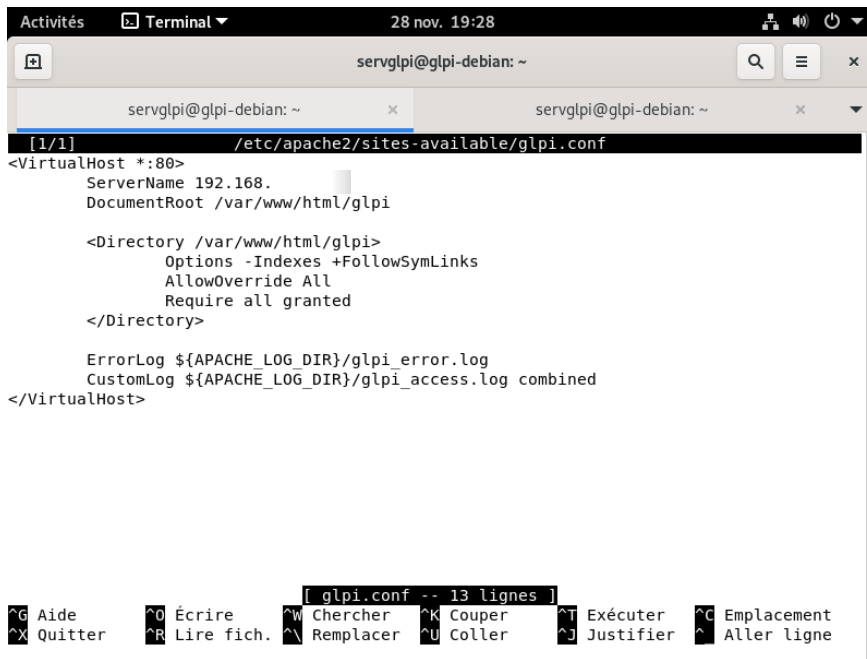
« **chown -R www-data:www-data** » change le propriétaire et le groupe de tous les fichiers du dossier **glpi** pour **www-data** (utilisateur apache).

« **chmod -R 755** » définit les permissions des fichiers et dossiers, « **7** » correspond à la lecture, l'écriture et l'exécution pour le propriétaire (**www-data**), « **5** » correspond à la lecture et l'exécution pour le groupe et les autres, « **-R** » applique la commande de manière récursive à tout le dossier et ses sous-dossiers.

Étape 9 : Configuration Apache

Après l'installation de GLPI dans `/var/www/html/glpi`, il faut configurer un VirtualHost dans Apache afin de rendre l'application accessible via un navigateur et garantir la sécurité et la bonne gestion des logs.

Pour créer le fichier rien de plus simple que « **nano /etc/apache2/sites-available/glpi.conf**. Je configure ensuite dans le fichier cette commande.



```
[1/1] /etc/apache2/sites-available/glpi.conf
<VirtualHost *:80>
  ServerName 192.168.
  DocumentRoot /var/www/html/glpi

  <Directory /var/www/html/glpi>
    Options -Indexes +FollowSymLinks
    AllowOverride All
    Require all granted
  </Directory>

  ErrorLog ${APACHE_LOG_DIR}/glpi_error.log
  CustomLog ${APACHE_LOG_DIR}/glpi_access.log combined
</VirtualHost>
```

- « **VirtualHost *:80** » déclare un site accessible sur tous les IPs de la machine sur le port 80(HTTP).
- « **ServerName 192.168.x.xxx** » définit le nom ou l'IP du serveur pour accéder au site.
- « **DocumentRoot /var/www/html/glpi** » est le répertoire où se trouvent les fichiers GLPI,
- « **<Directory /var/www/html/glpi>** » est le bloc de configuration spécifique au dossier GLPI
- « **Options -Indexes +FollowSymLinks** » « **-Indexes** » désactive la liste des fichiers si index absent
- « **+FollowSymLinks** » permet de suivre les liens symboliques.
- « **AllowOverride All** » permet à GLPI d'utiliser les fichiers.
- **.htaccess** pour gérer certaines configurations
- « **Require all granted** » autorise l'accès à tous les utilisateurs
- « **ErrorLog \${APACHE_LOG_DIR}/glpi_error.log** » est le fichier de logs erreurs du site
- « **CustomLog \${APACHE_LOG_DIR}/glpi_access.log combined** » est le fichier de logs des accès HTTP avec format combiné.

Pour enregistrer la configuration : CTRL+O / ENTRÉE / CTRL+X

Un VirtualHost apache a été configuré pour GLPI afin de définir l'IP du serveur, le répertoire web, les permissions d'accès et les fichiers de logs. Cette configuration permet à glpi de fonctionner correctement, gérer ses propres fichiers et d'assurer la sécurité et suivi des accès via des logs dédiés.

9.2 Activation du site

Après avoir créé le fichier de configuration **glpi.conf** pour Apache, il faut activer le site, le module requis et redémarrer Apache. Sur certaines installations Debian comme la mienne, les commandes « **a2ensite** » et « **a2enmod** » ne sont pas dans le **PATH** par défaut et doivent être appelées via leurs chemin absolu.

Pour activer le site j'utilise donc la commande « **/usr/sbin/a2ensitre glpi.con** »

« **/usr/sbin/a2ensitre** » est le chemin absolu vers la commande a2ensite, « **glpi.con** » est le fichier de configuration créé pour GLPI. La commande permet de créer un lien dans **/etc/apache2/sites-enabled/**.

Une fois fait, j'active le module **rewrite** « **/usr/sbin/a2enmod rewrite** »

« **/usr/sbin/a2enmod** » est le chemin absolu vers la commande a2enmod, « **rewrite** » est le module permettant la réécriture d'URL, nécessaire pour GLPI afin d'utiliser ses URLs propres et fonctionnalités **.htaccess**.

Pour que toutes les modifications soient prises en compte, il faut redémarrer Apache2 avec la commande « **systemctl restart apache2** » une fois que le Apache est redémarré, le site GLPI est actif et directement accessible depuis l'IP définie dans le VirtualHost.

Étape 10 : Installation via l'interface web

10.1 Choix de la langue



The screenshot shows the 'GLPI SETUP' interface. At the top left is the GLPI logo. To its right, the text 'GLPI SETUP' is displayed in orange. Below this, the instruction 'Sélectionnez votre langue' is centered. A white dropdown menu is shown with 'Français' selected. At the bottom left, there is an orange button labeled 'OK >'.

10.2 Connexion à la base de donnée crée précédemment



The screenshot shows the 'GLPI SETUP' interface at 'Étape 2'. The title 'GLPI SETUP' is in orange. Below it, 'Étape 2' is centered, followed by the subtitle 'Test de connexion à la base de données'. A green success message is displayed in a white box: '✓ Connexion à la base de données réussie'. Below this, the instruction 'Veuillez sélectionner une base de données à mettre à jour :' is shown. A selection box contains a radio button next to the text 'glpidb'. At the bottom left, there is an orange button labeled 'Continuer >'.

10.3 Choisir l'installation de la base de données



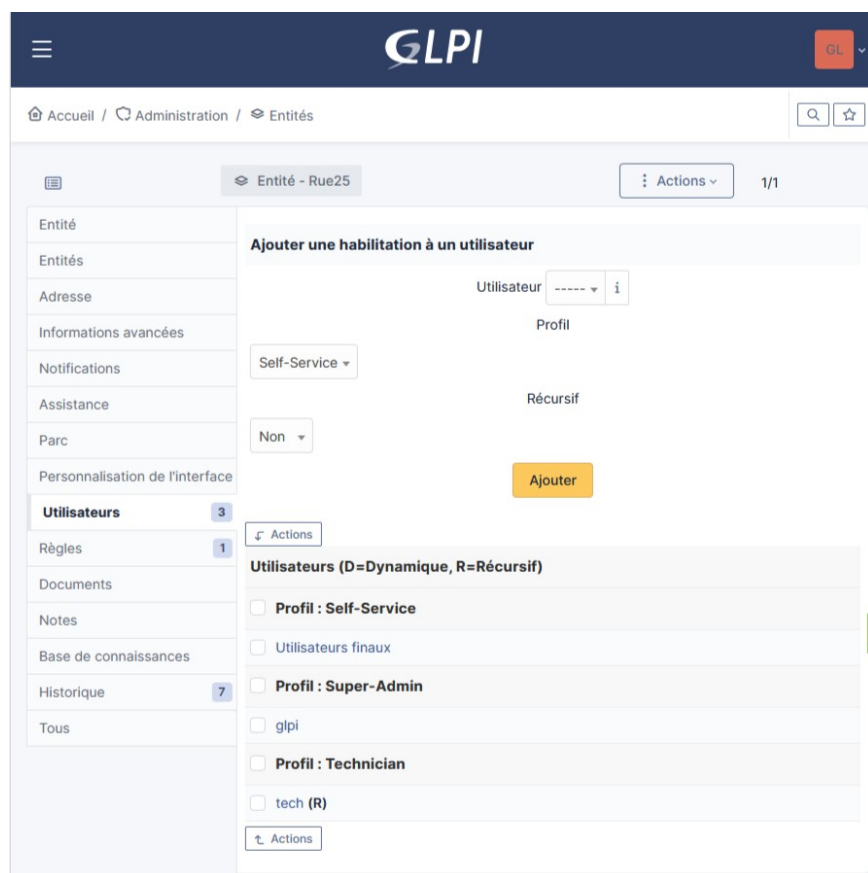
10.4 Récolte de données



10.5 Fin de l'installation



Il ne reste plus qu'à faire les configurations en fonction de l'entité gérée, du parc d'appareils, des utilisateurs etc.



GLPI

GL

Accueil

Administration

Entités

Entité - Rue25

Actions

1/1

Entité

Entités

Adresse

Informations avancées

Notifications

Assistance

Parc

Personnalisation de l'interface

Utilisateurs 1

Règles 1

Documents

Notes

Base de connaissances

Historique 6

Tous

Nom

Rue25

Commentaires

Sauvegarder

Utilisateur

Habilitations 1

Groupes

Préférences

Éléments utilisés

Éléments gérés

Tickets créés

Problèmes

Changements

Documents

Réservations

Synchronisation

Liens

Certificats

Historique

Tous

Identifiant

glpi

Image

GL

Fichier(s) (2 Mio maximum)

Glissez et déposez votre fichier ici, ou

Choisir un fichier

Aucun fichier choisi

Effacer

Nom de famille

Prénom

Mot de passe

Confirmation mot de passe

Fuseau horaire

L'utilisation des fuseaux horaires n'a pas été activé. Exécutez la commande "php bin/console database:enable_timezones" pour l'activer.

Actif

Oui

Courriels

Valide depuis

Sources :

Installation complète de GLPI 10 sur Debian

- GLPI installation configuration : <https://www.it-connect.fr/installation-pas-a-pas-de-glpi-10-sur-debian-12/>
- Installation GLPI 10.x sur ma VM Debian : <https://neptunet.fr/install-glpi10/>
- Configuration Apache pour GLPI : <https://www.dsfc.net/parc/glpi/configuration-apache-glpi/>
- Document PDF d'installation GLPI : <https://alcorp.info/assets/img/Documents/GLPI.pdf>

Gestion des utilisateurs, profils et habilitation dans GLPI

- Documentation officielle GLPI – Profils et habilitations : https://help.glpi-project.org/faq/fr/glpi/profile_habilitation
- Forum GLPI, gestion des entités et délégation des droits : <https://forum.glpi-project.org/viewtopic.php?id=282708>
- WikiGite, entité et utilisateurs dans GLPI : https://wiki.kogite.fr/index.php/Les_entit%C3%A9s_et_les_utilisateurs

Test réseau : ping, SSH, connectivité

- Top 15 commandes réseau Linux : <https://www.it-connect.fr/top-15-des-commandes-linux-pour-administrer-et-diagnostiquer-le-reseau/>
- Guide complet des commandes réseau Linux : <https://blog.stephane-robert.info/docs/admin-serveurs/linux/reseaux/>
- Commande SSH, ping, FTP, Telnet : <https://www.guru99.com/fr/communication-in-linux.html>

Projet : Infrastructure IT – Agence Rue25 (fictive)

Auteur : IJustStartPython

Année : 2025

Usage : Portfolio professionnel