

Création d'un serveur maison

Objectifs du projet

- Centraliser sur un serveur les photos et vidéos dispersées sur 3 ordinateurs différents.
- Pouvoir accéder aux données depuis une machine Windows connectée au réseau Wi-Fi.

Problèmes initiaux

- Les données sont dispersées sur trois ordinateurs différents, ce qui est peu pratique.
- Les systèmes d'exploitation des ordinateurs se font vieux.
- Les données n'ont pas de backup, donc si un disque dur tombe en panne elles sont perdues.

Aperçu des PC

Voici un aperçu des 3 PC, avec leurs systèmes d'exploitation et leurs disques durs respectifs :

- PC HP de 2009, sous Windows Vista, avec un HDD de 640 Go
- PC Acer de 2011, sous Windows 7, avec un HDD de 1.5 To
- PC gaming Asus de 2017, sous Windows 10, avec un SSHD de 1 To

Au total, cela fait un peu plus de 3 To de stockage.

Disques durs

D'après le logiciel Crystal Disk Info, les 3 disques durs sont dans un état "bon" malgré leur âge avancé. Même s'ils étaient dans un état mauvais, ce n'aurait pas été si grave : leur contenu sera des backups donc si un disque dur cesse de fonctionner, une copie de son contenu est stockée autre part.

De plus, un disque dur est assez coûteux et il serait inutile et peu écologique de jeter des composants qui fonctionnent encore. J'ai donc décidé de réutiliser ces 3 disques durs pour mon serveur, en les mettant dans le PC le moins ancien (PC gaming Asus de 2017).



Figure 1 : Disques durs installés dans le PC

Système d'exploitation

Le PC démarre très lentement, et son OS n'a pas été réinstallé depuis 8 ans. Il est donc nécessaire de le changer, pour repartir sur une installation propre.

Au départ, je voulais passer sous Windows 11, mais la machine n'est pas compatible. Le support de Windows 10 arrivant à sa fin quelques mois après le projet, j'ai préféré prévenir plutôt que guérir, et j'ai décidé de passer sous Linux.

Après des recherches sur Reddit, mon choix de distribution s'est porté sur Debian. Je ne voulais pas une distribution spéciale serveur au cas où j'ai besoin d'effectuer d'autres tâches sur l'ordinateur, j'ai donc opté pour une distribution généraliste. Debian correspond à ce critère, et est en plus assez légère, ce qui rend le démarrage du serveur encore plus rapide. De plus, c'est une distribution que je n'avais jamais utilisée avant ce projet, cela m'a donc permis d'en découvrir une en plus.

Configuration

Le SSHD de 1 To étant le disque plus récent et le plus rapide, c'est sur celui-ci que j'ai installé l'OS.

Au niveau de l'interface graphique, celle de base, GNOME, était installée avec beaucoup de logiciels inutiles. J'avais seulement besoin d'un explorateur de fichiers et d'un navigateur internet, je me suis donc porté sur Xfce qui propose ces deux outils tout en étant très léger.

Pour monter les deux autres disques durs sous Debian, j'ai dû modifier le fichier `/etc/fstab` afin qu'ils aient un dossier chacun dans le dossier `/srv/sauvegardes/`.

J'ai créé deux comptes sur la machine : un compte administrateur, qui peut modifier tout le contenu du serveur, et un compte utilisateur classique qui peut seulement lire le contenu du dossier sauvegardes, mais qui peut cependant modifier le dossier `vac`.

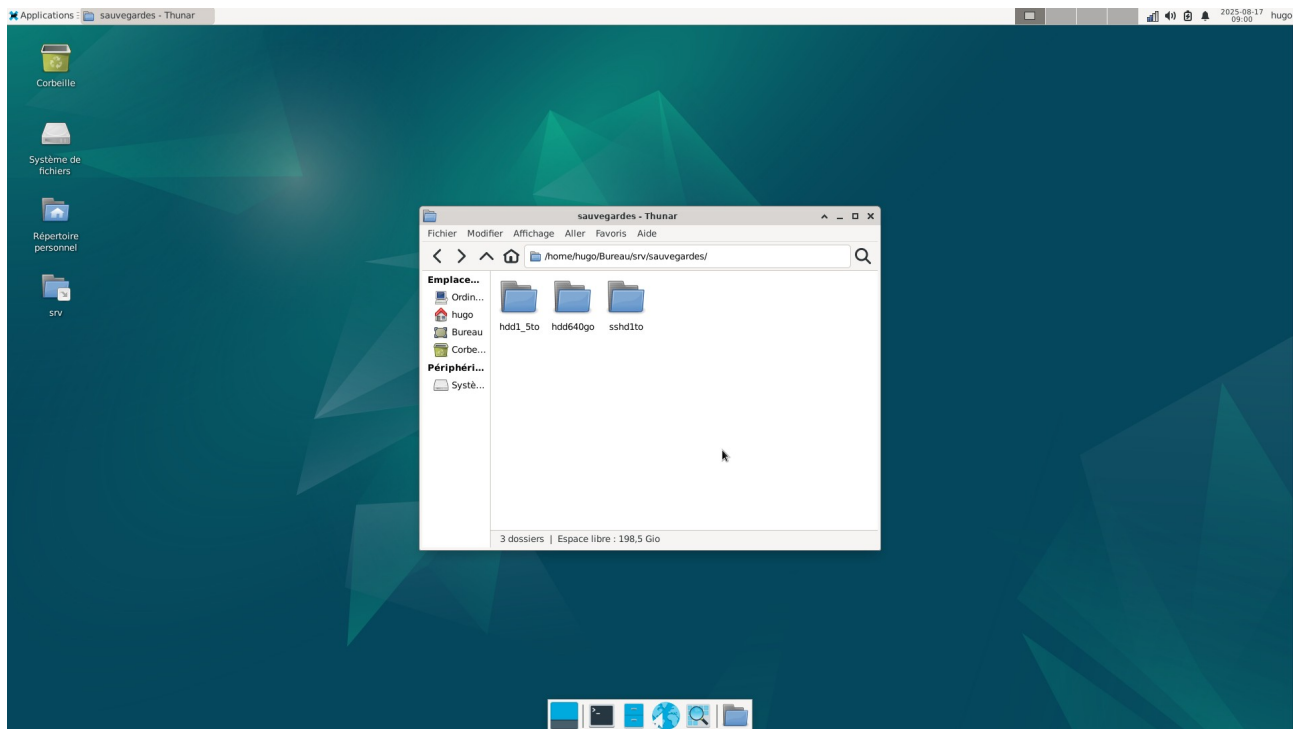


Figure 2 : Page d'accueil du serveur

Accès à distance

Je voulais pouvoir visionner les photos et vidéos stockées sur le serveur depuis mon ordinateur portable, qui est sous Windows 11. Pour ce faire, j'ai créé un partage SMB grâce au package `samba`. Après avoir modifié le fichier de configuration, les backups sont accessibles depuis n'importe quel ordinateur connecté au Wi-Fi, à condition d'avoir un identifiant et un mot de passe valides.

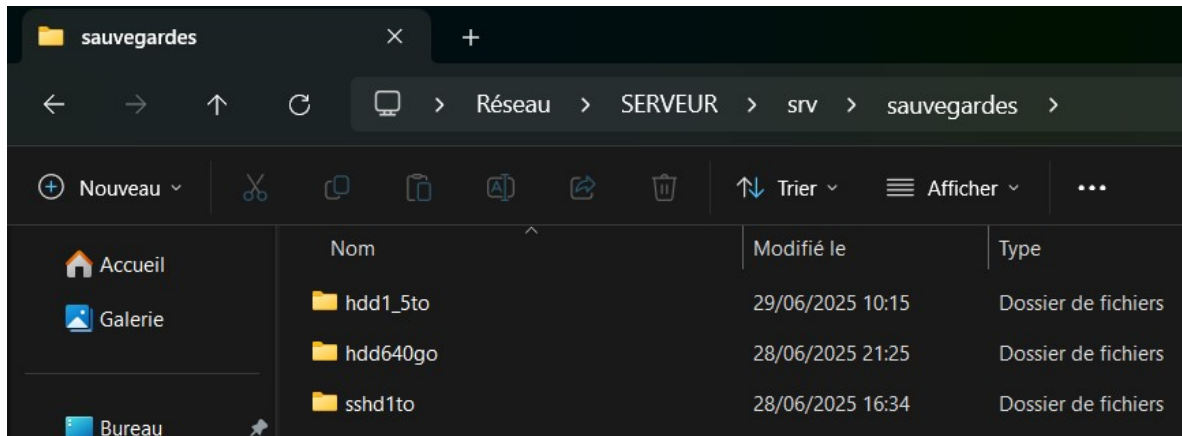


Figure 3 : Dossiers du serveur accessibles depuis un PC Windows connecté au Wi-Fi

Résultat final

Au final, j'ai réussi à :

- Monter deux disques durs supplémentaires sur un PC
- Formater puis partitionner ces disques durs
- Installer un système d'exploitation Linux (Debian)
- Gérer les permissions du serveur
- Configurer un partage de fichiers SMB, protégé par un mot de passe, pour accéder aux fichiers depuis le Wi-Fi