TP 9
Sauvegarde

Sommaire

Introduction 2
I/ Windows 2
1) Créer un dossier partagé pour votre voisin 2
2) Sauvegarde Windows 6
3) Logiciel Duplicati 11
4) Script bat (planification) 18
II/ Linux 23
1) Partage NFS 23
2) Utiliser Duplicati 26
3) Sauvegarde manuelle (cp, tar, at, crontab) 36
1. Commande cp 36
2. Commande tar 36
3. Planification avec at 37
4. Planification avec Crontab 37
4) Tester la distribution RedoBackup 38
5) Synchronisation avec Unison 50
1) Linux 50
2) Windows 60
3) Windows / Linux 67

Introduction

Questions préalables :

- Que faut-il sauvegarder et à quelle fréquence ? Données chaudes/froides
- Sur quels supports ? NAS, Disque, Cloud, Clé
- Que faire en cas d'incident / accident ? PCA/PRA
- Mes données valent-elles la peine d'être sauvegardées

Première source de perte données : l'erreur humaine

- Synchro pour les données brûlantes
- Sauvegarde journalière (x6), hebdomadaire (x4), mensuelle (x12)

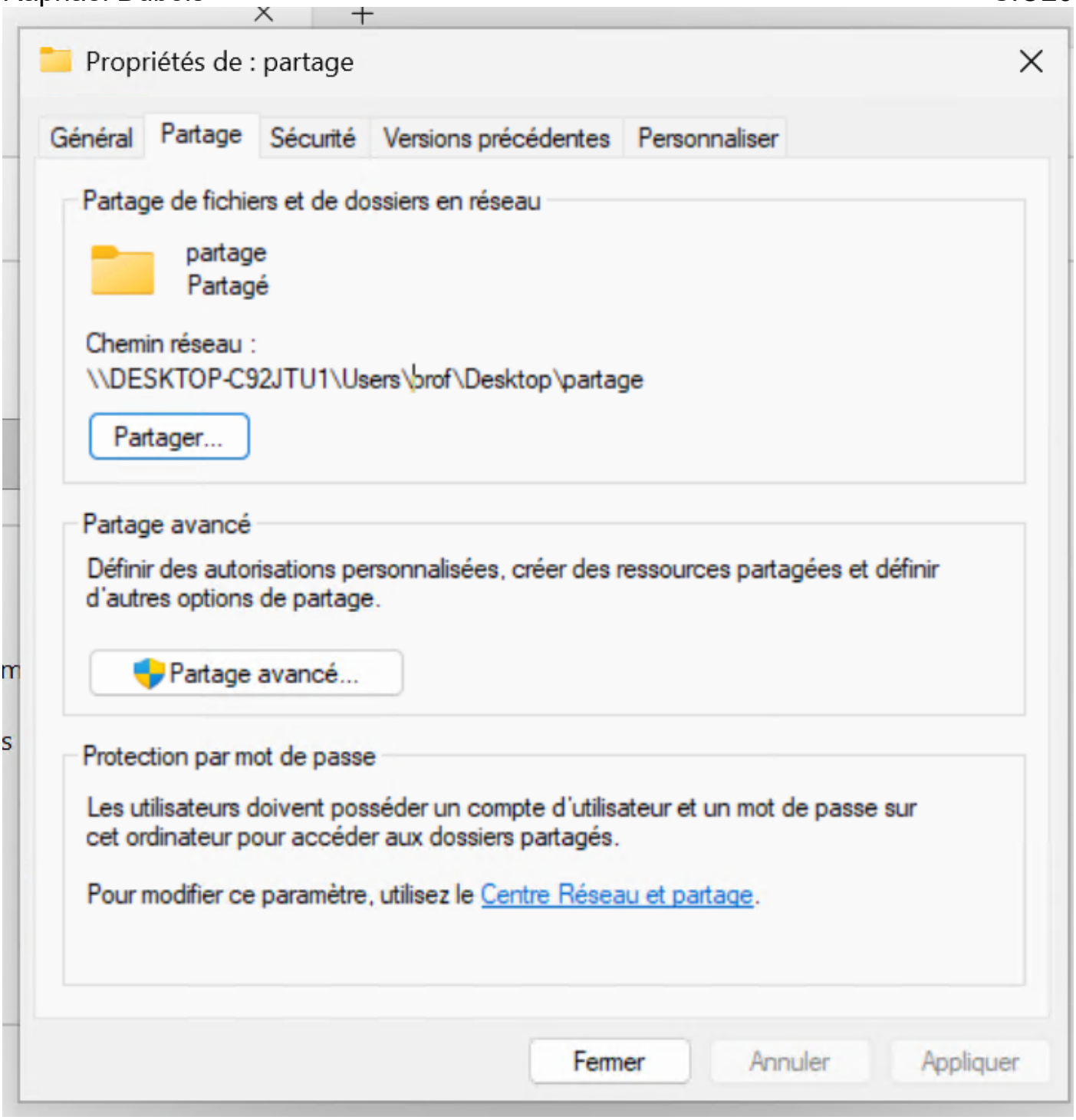
Différents types de sauvegarde :

- Totale (complet mais volumineux)
- Différentielle (moins volumineux mais plus long à reconstituer)
- Incrémentale (moins volumineux et plus rapide, conserve les différentes versions)

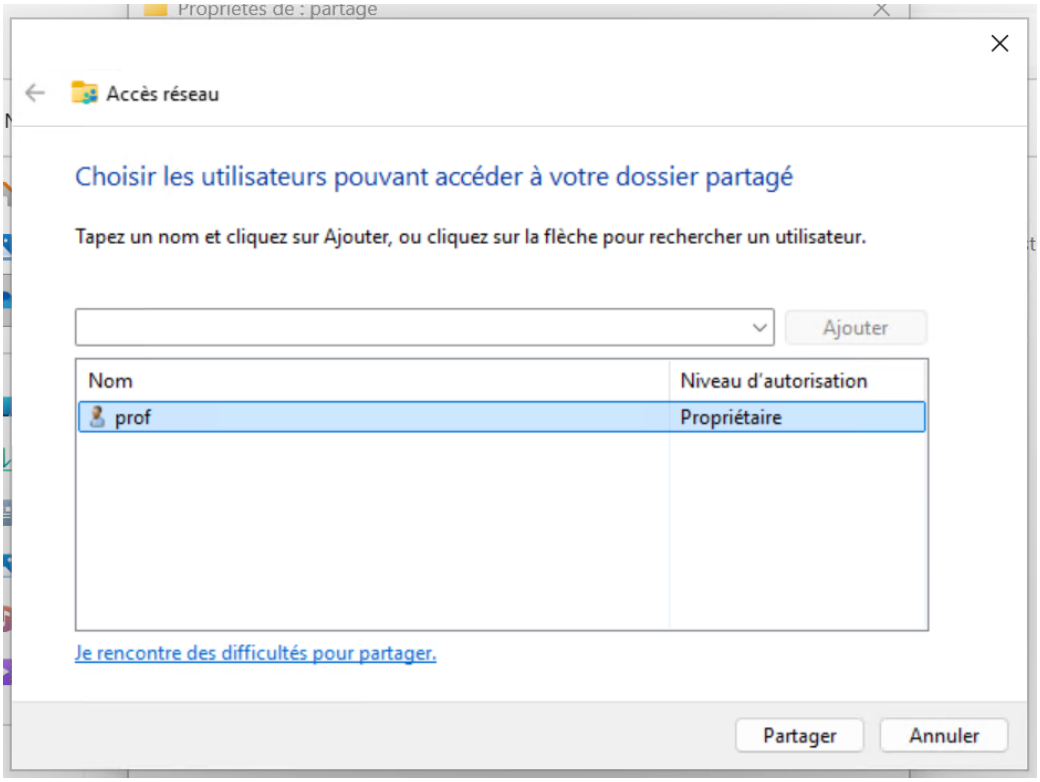
I/ Windows

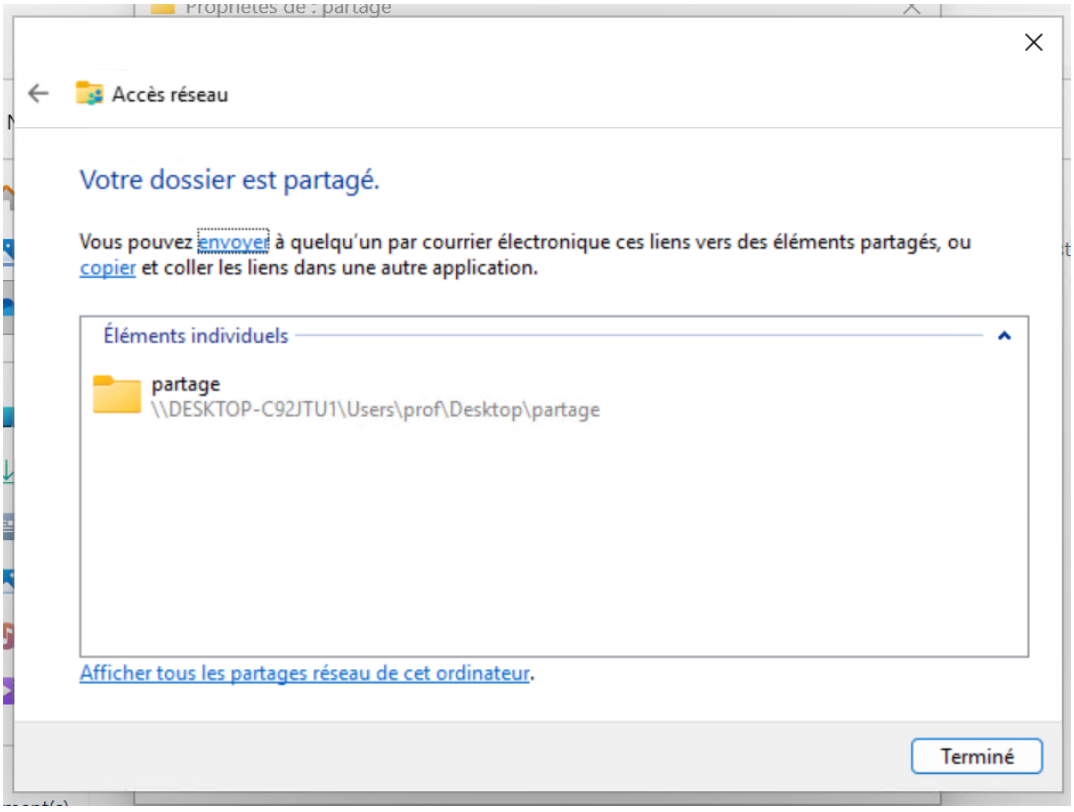
1) Créer un dossier partagé pour votre voisin

Pour créer un dossier sous Windows, il suffit de faire clic droit sur le dossier que nous souhaitons partager à l'ordinateur distant, puis propriété, et enfin il faut se rendre dans l'onglet « Partage » :



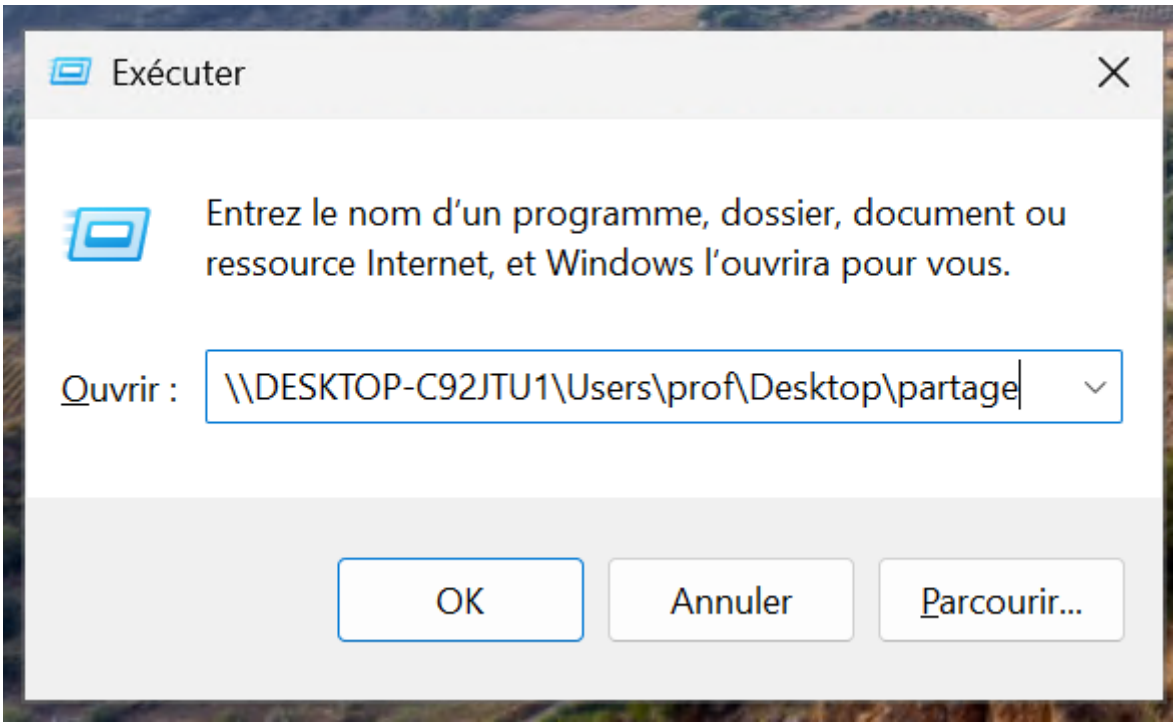
Une fois sur cet écran, nous pouvons cliquer sur « partager » et enfin, sélectionner l'utilisation pouvant accéder au dossier partager :

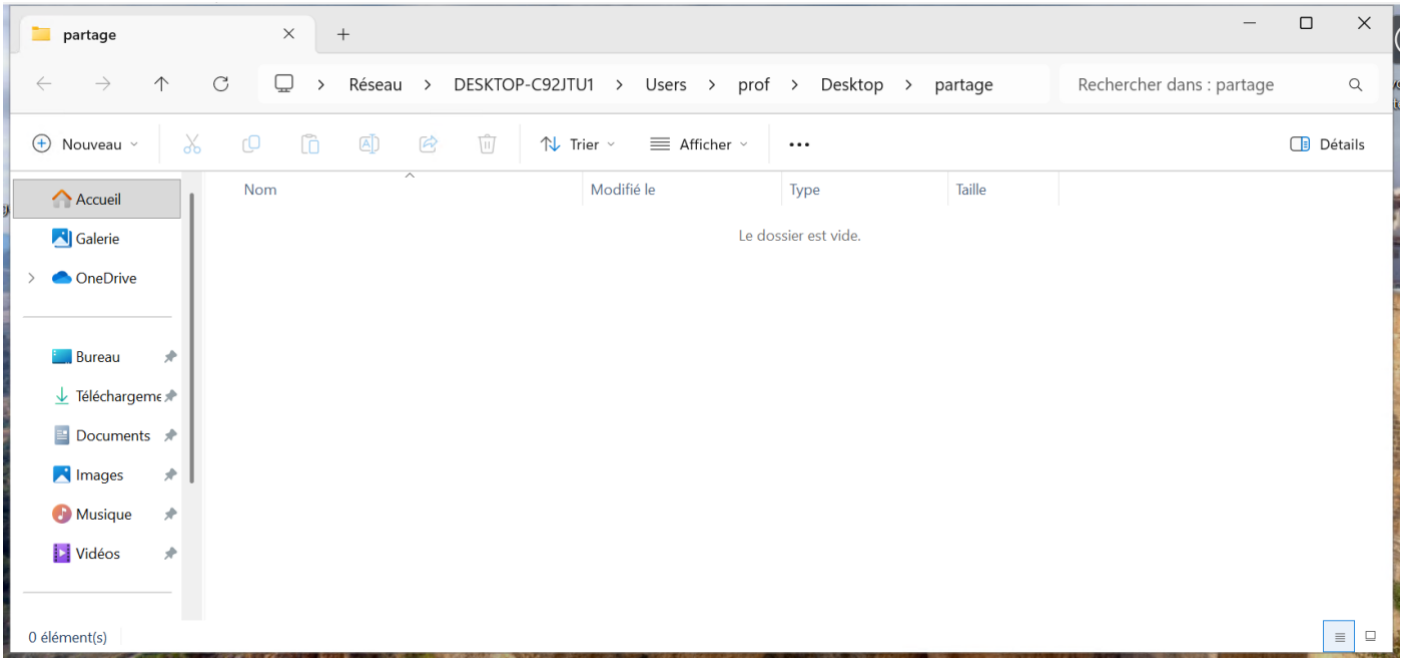




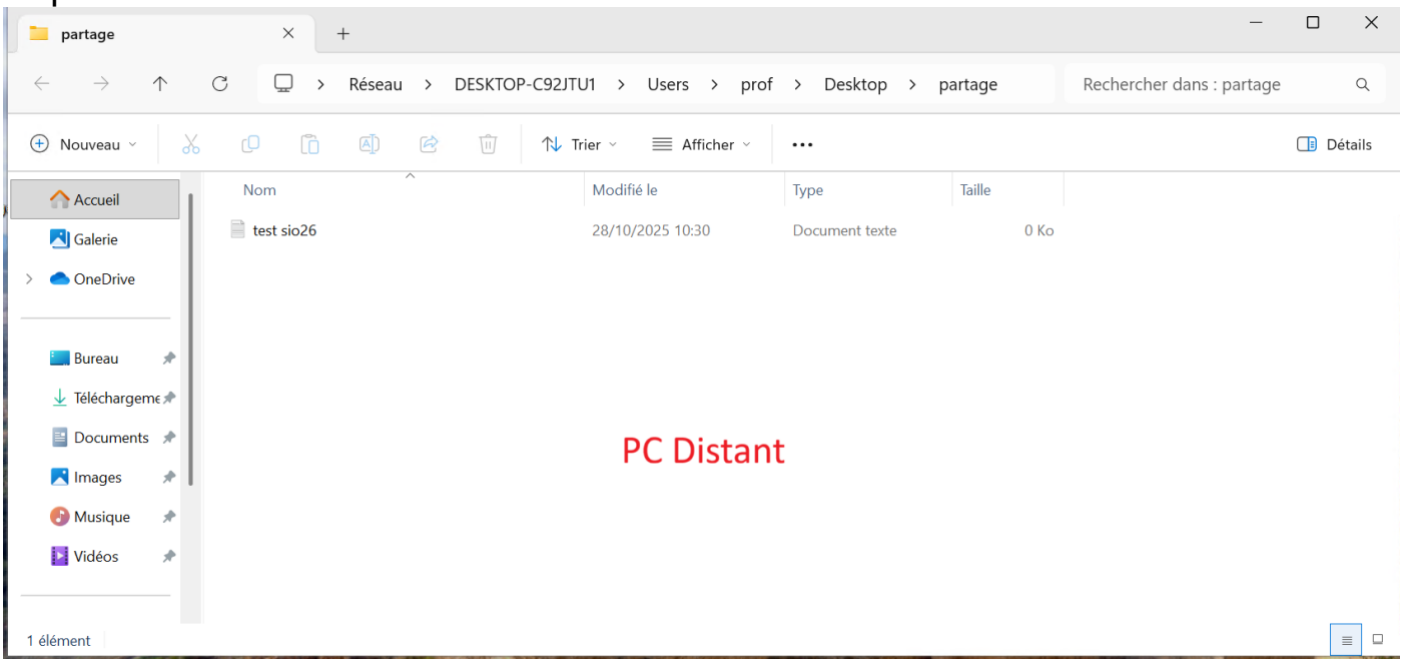
Une fois ceci fait, il suffit de récupérer le chemin \\DESKTOP-C92JTU1\Users\prof\Desktop\partage et d'y accéder via le PC distant. Pour cela, il

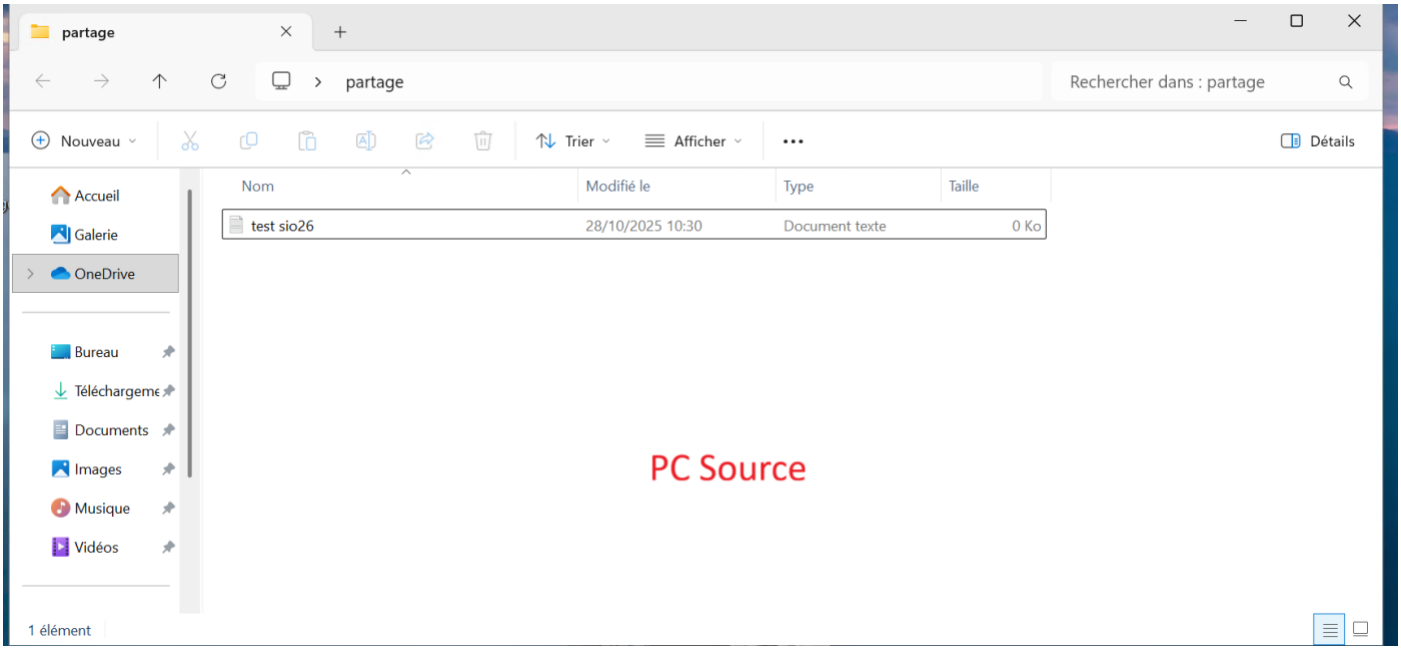
suffit d'ouvrir la fenêtre exécuter grâce à Windows + R et de coller le chemin distant sur le PC distant :





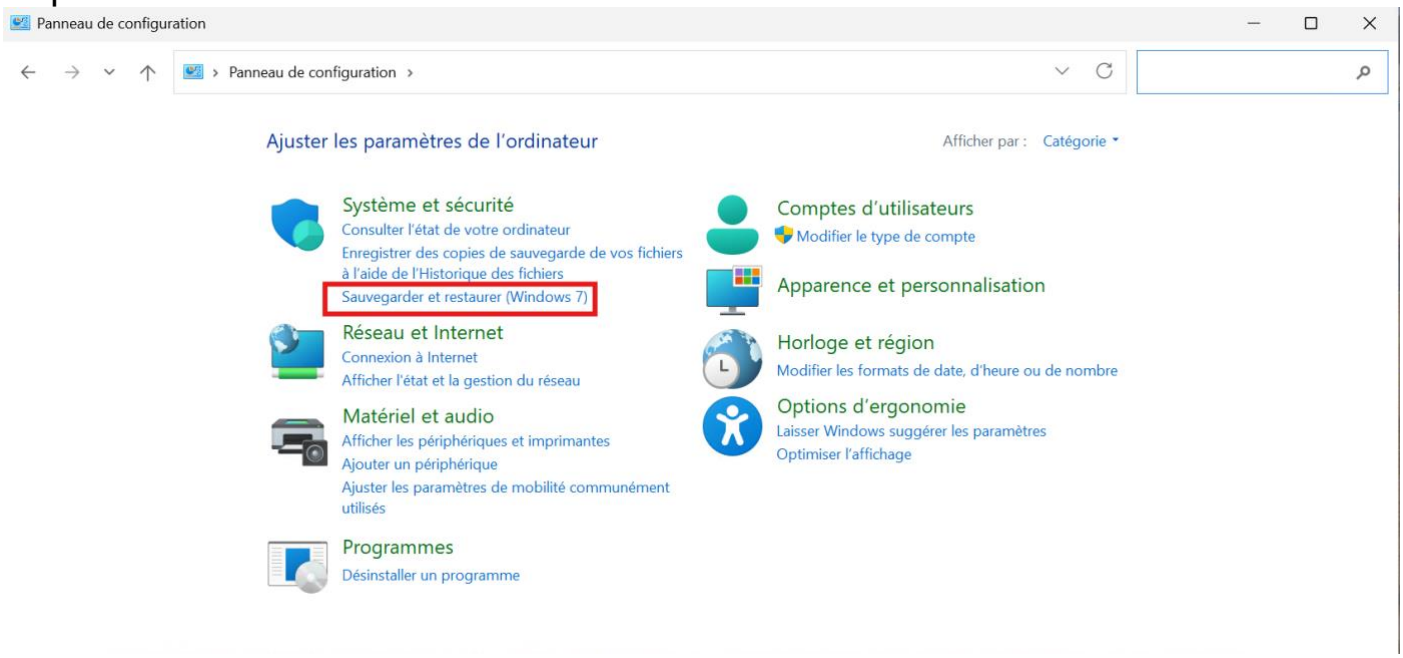
Une fois ceci fait, nous avons accès au dossier distant ! Pour vérifier que cela fonctionne, nous pouvons créer un fichier texte et vérifier qu'il apparait bien sur l'autre PC.

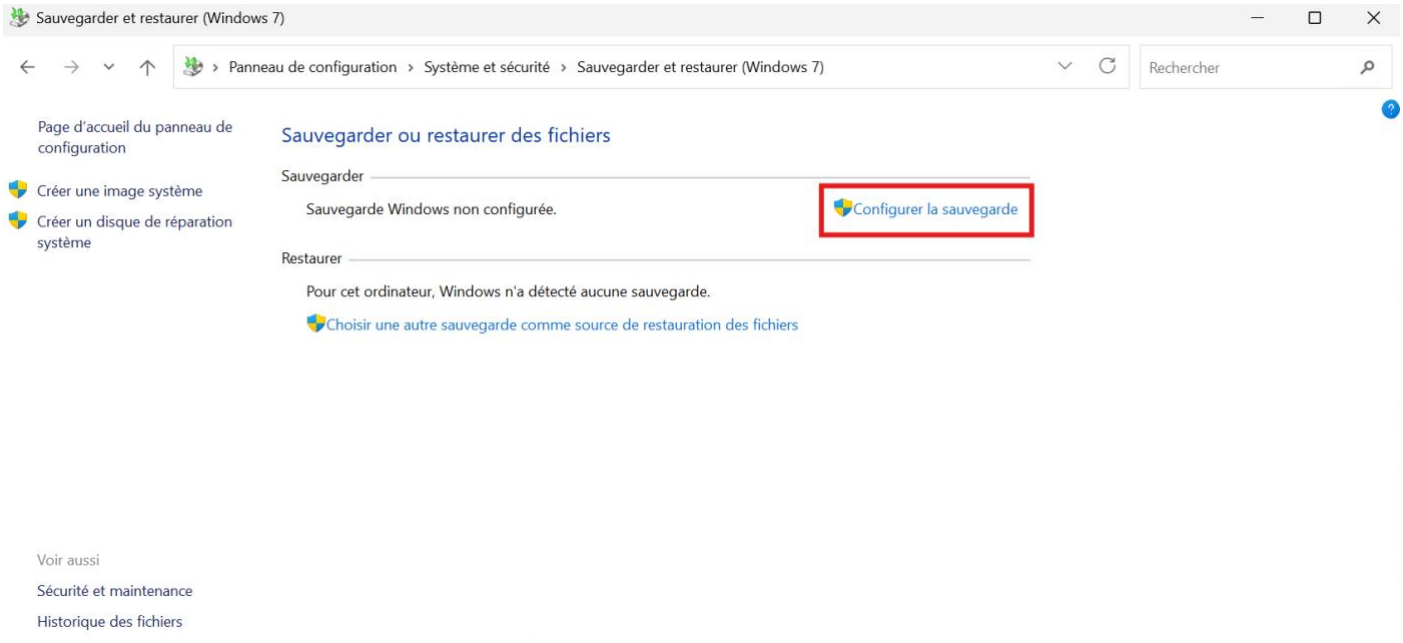




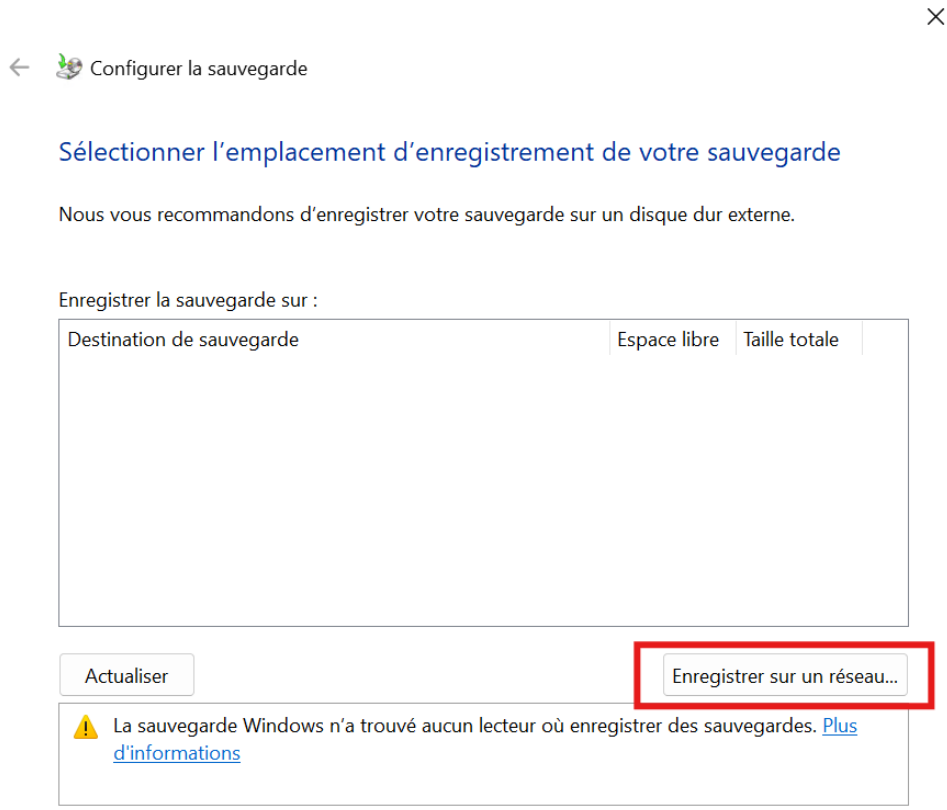
2) Sauvegarde Windows

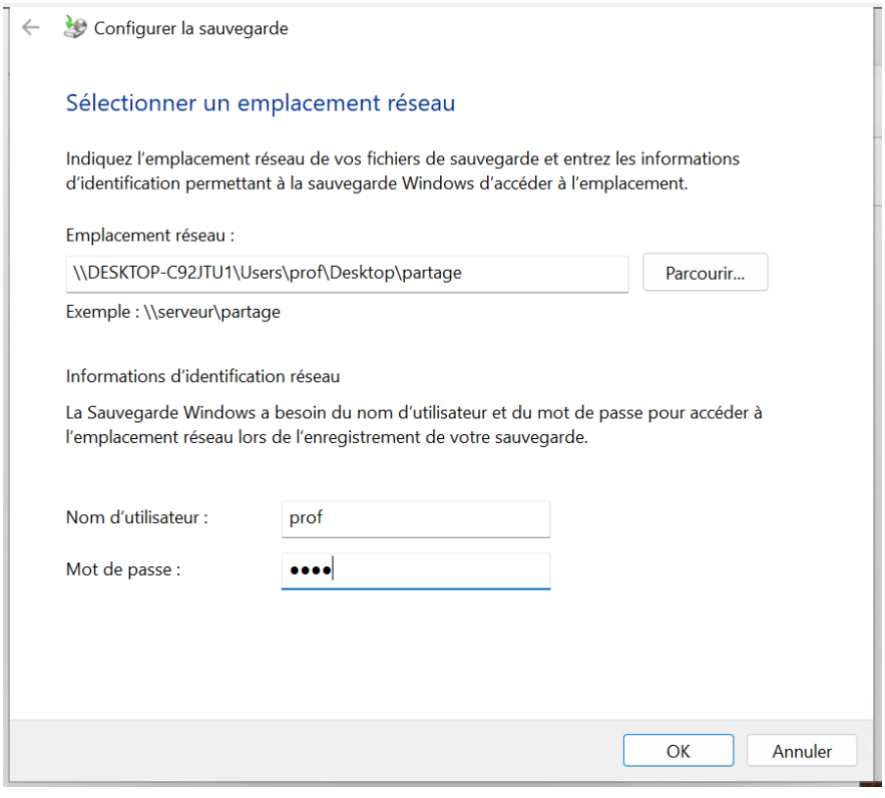
Sur le PC distant, nous allons lancer l'outil de sauvegarde Windows pour sauvegarder nos données vers le PC source. Pour cela nous allons nous rendre dans le panneau de configuration Windows, puis « Sauvegarde et restaurer (Windows 7) » :





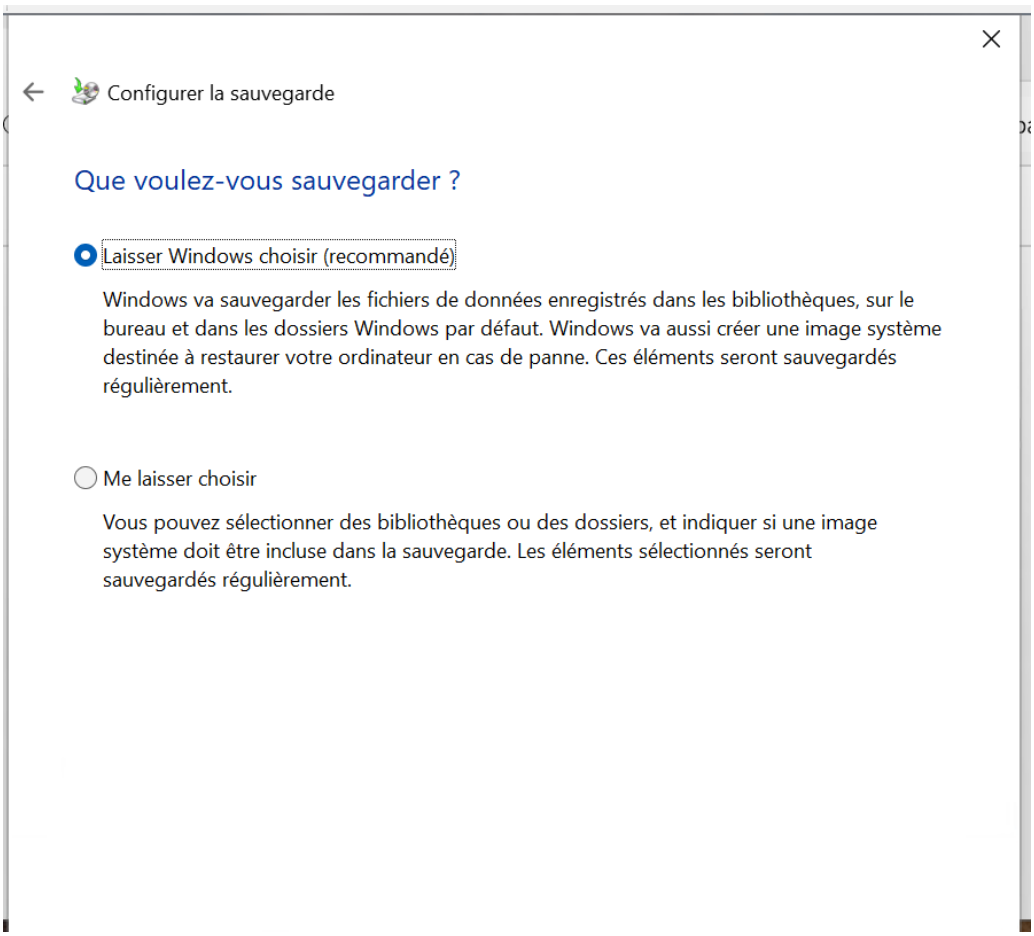
Sur la prochaine fenêtre, nous allons cliquer sur « Enregistrer sur le réseau », puis nous allons indiquer le chemin avec le nom d'utilisateur et mot de passe pour que l'outil puisse se connecter au partage :



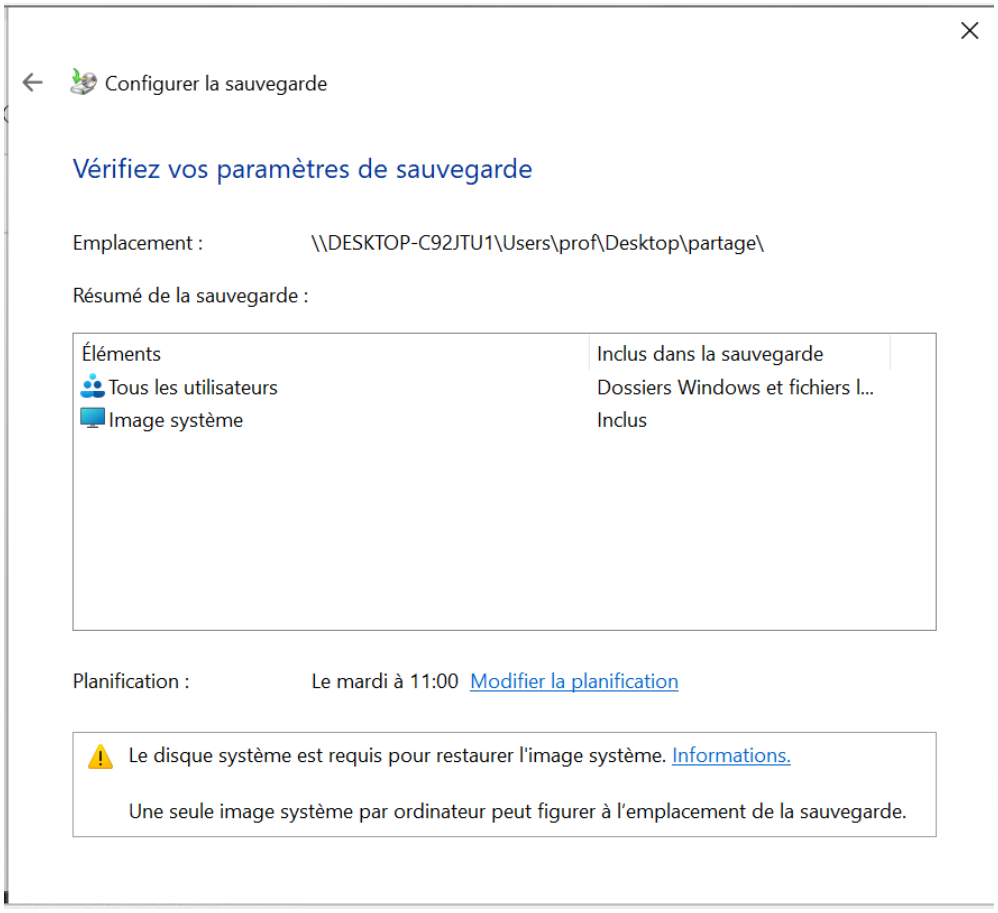


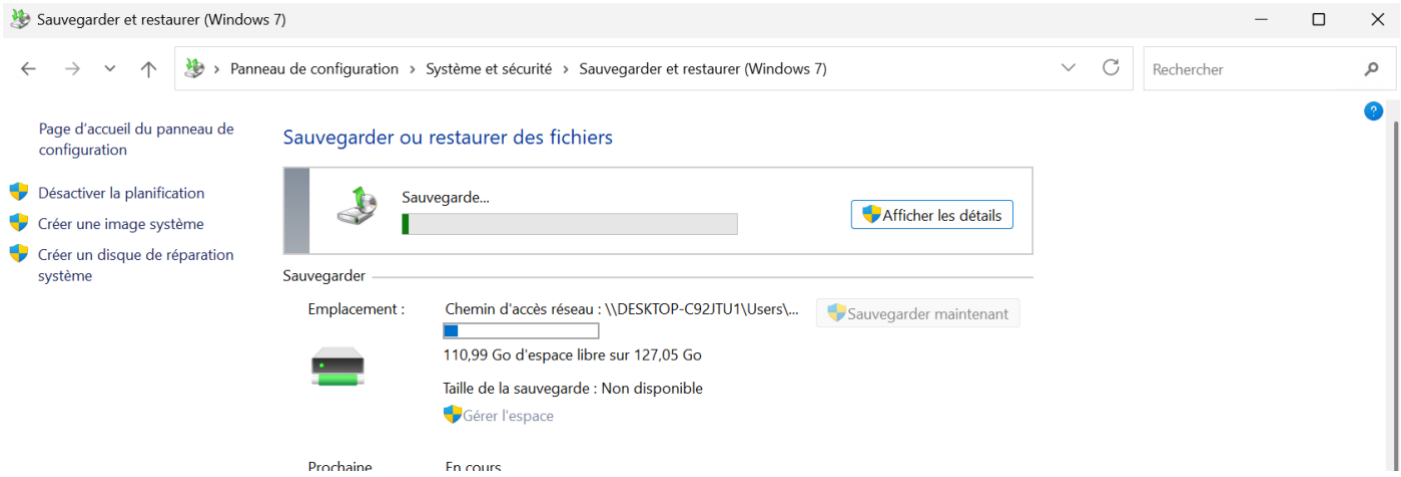
Une fois ceci fait, nous pouvons cliquer sur suivant, Windows va nous demander ce que nous souhaitons sauvegarder, dans notre cas nous allons laisser Windows choisir

ce qui est nécessaire à sauvegarder, mais il est aussi possible de sélectionner des données en particulier à sauvegarder :



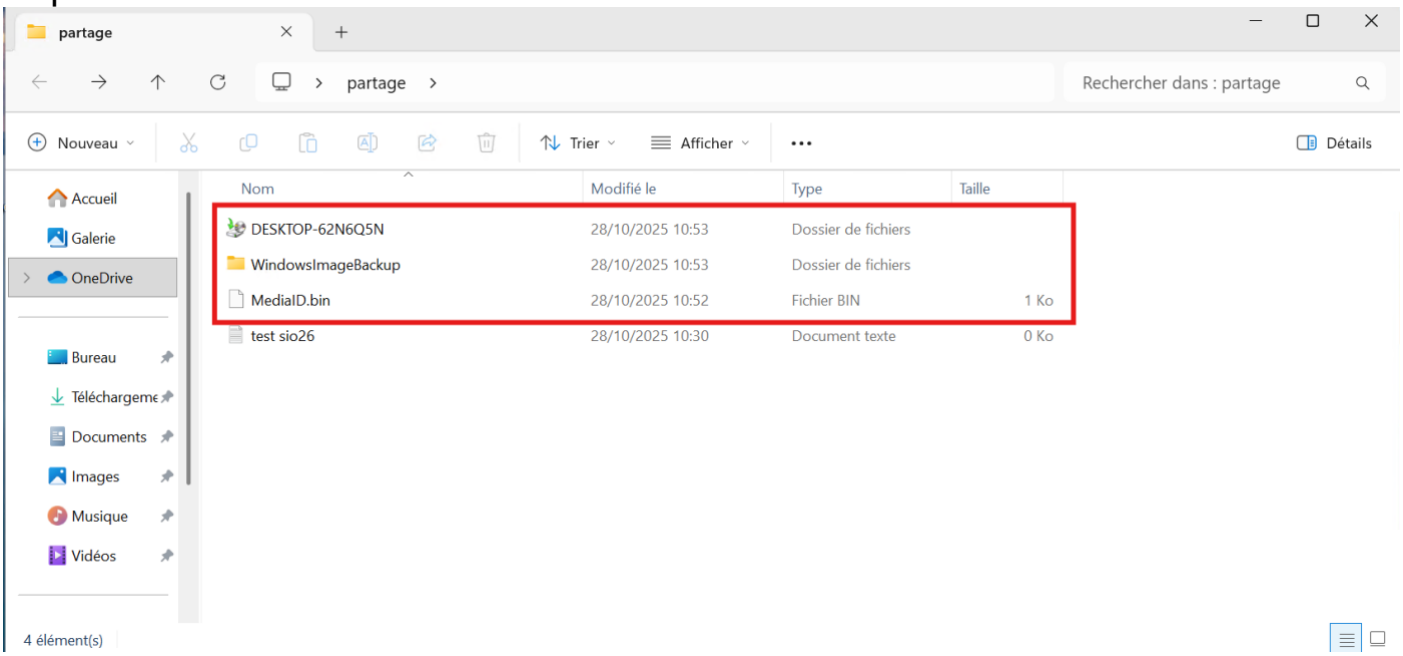
Une fois ceci fait, nous avons un récapitulatif des données qui vont être sauvegardé et nous pouvons continuer :





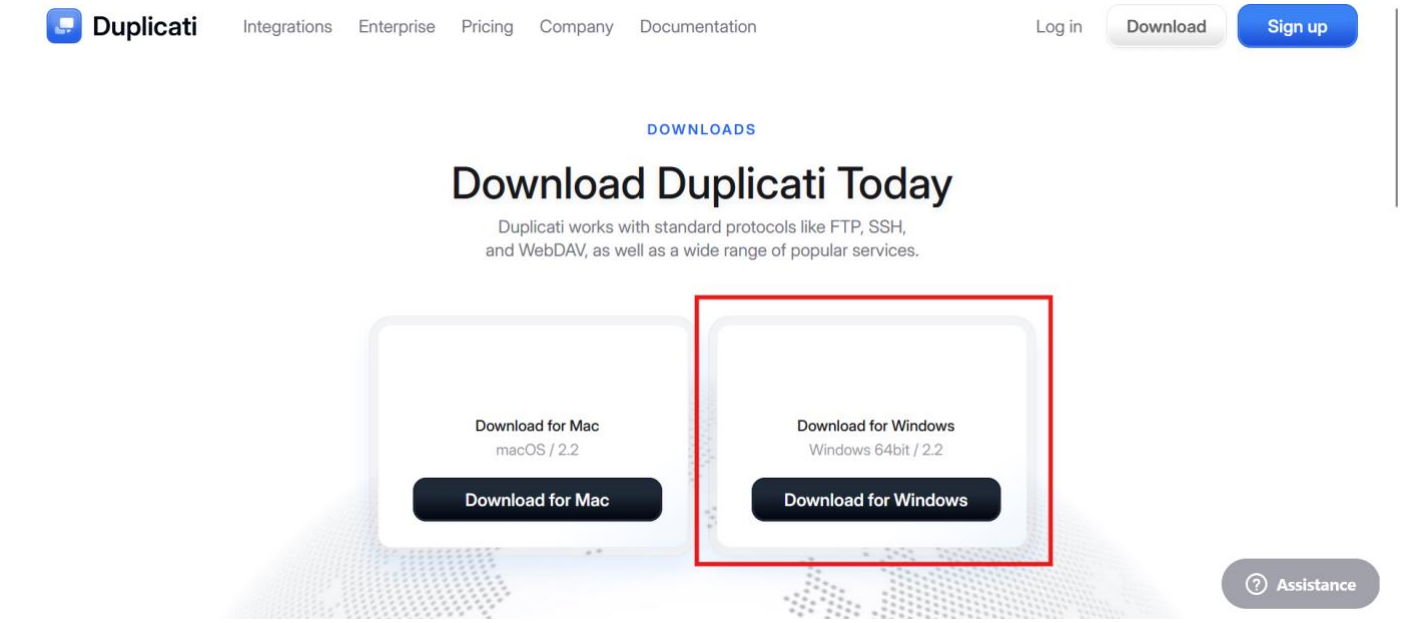
Une fois toutes les opérations réalisées, il suffit de patienter que le sauvegarde se termine.

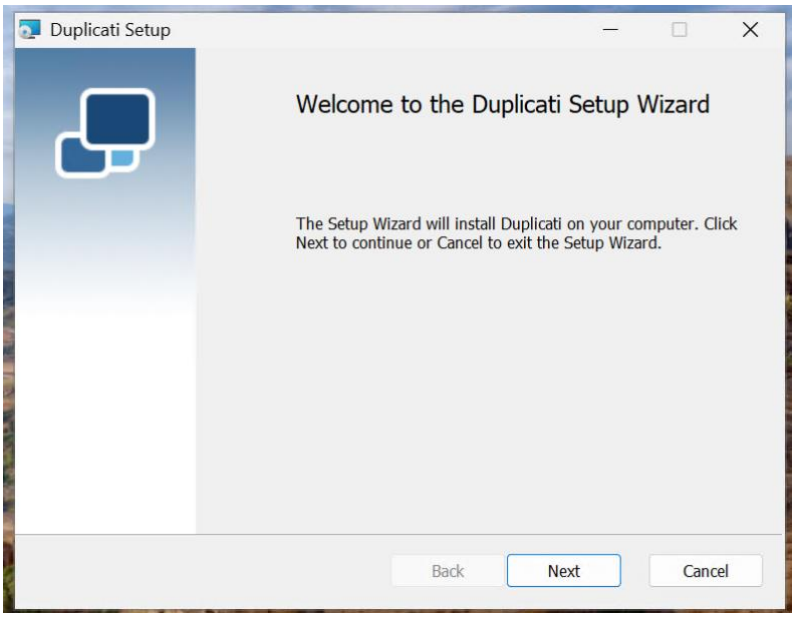
Une fois terminé, nous obtenons plusieurs dossiers et fichiers dans notre dossier partagé qui correspondent à la sauvegarde de notre PC distant :



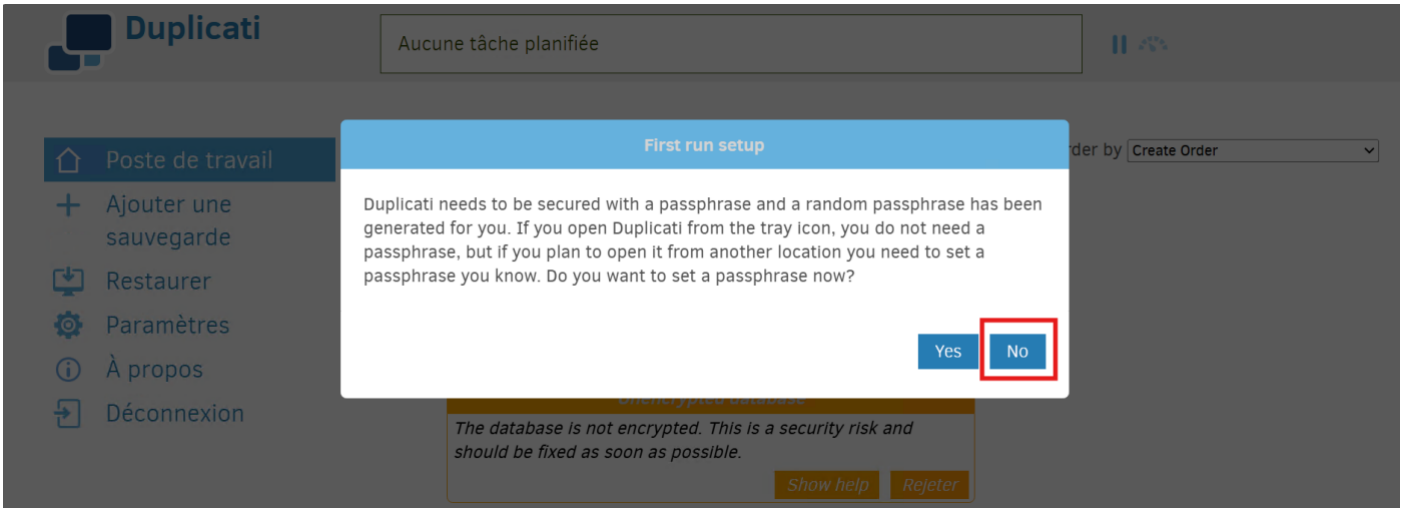
3) Logiciel Duplicati

Premièrement, nous devons télécharger et installer le Duplicati (https://duplicati.com/download) et l'installer sur notre PC distant :

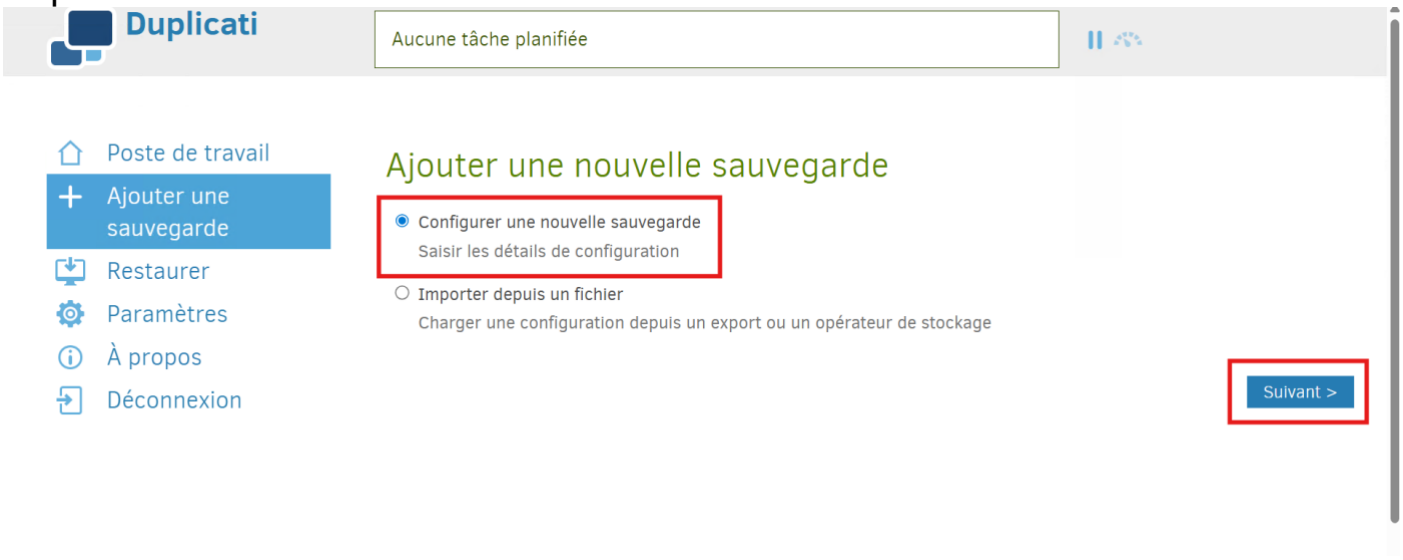


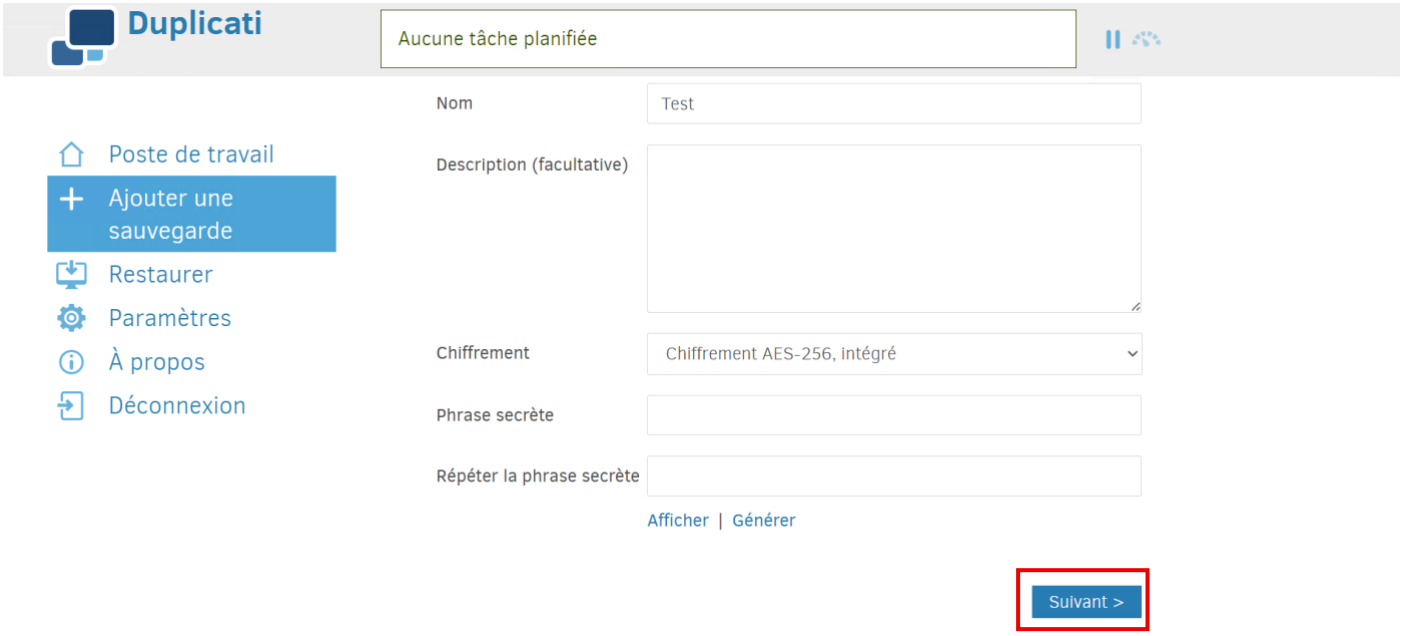


Une fois l'installation terminée, nous arrivons sur l'interface web de Duplicati ou nous allons pouvoir commencer à configurer, au premier démarrage, le logiciel va demander de définir une passphrase, dans notre cas, pour tester le logiciel, nous n'allons pas en définir pour le moment :



Nous allons maintenant nous rendre dans « Ajouter une sauvegarde » puis « Configurer une nouvelle sauvegarde » :

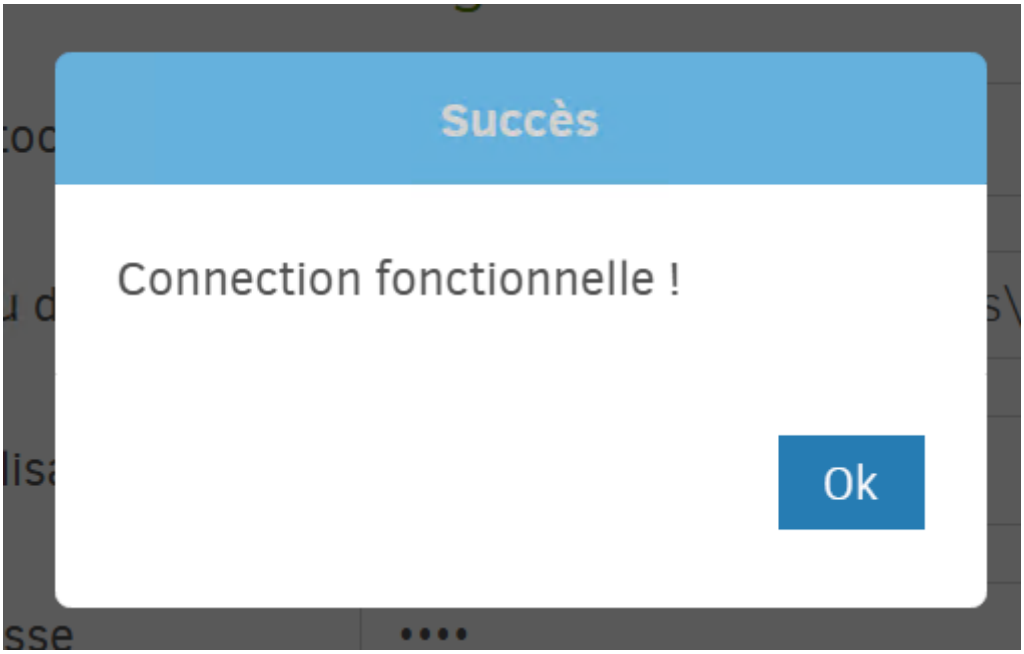




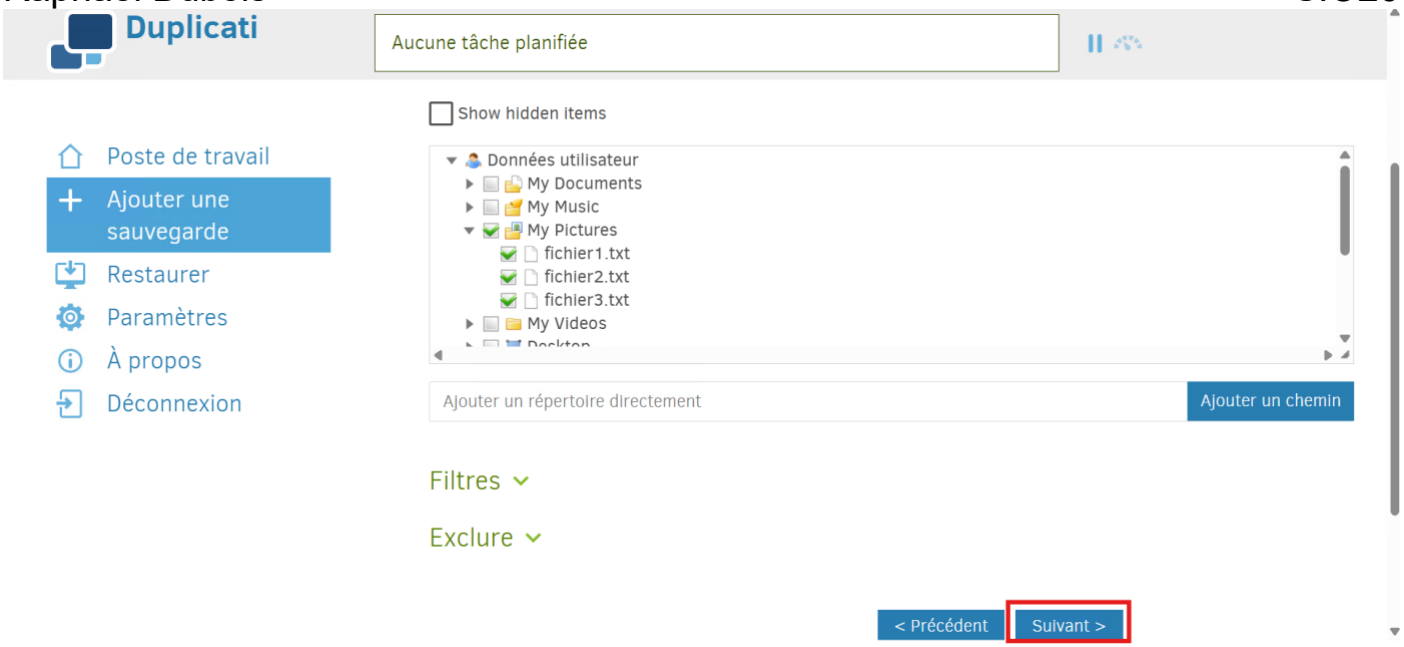
Ensuite, nous allons définir le chemin de destination dans Duplicati, pour cela, nous devons sélectionner le type de stockage en dossier ou disque local, définir le chemin sur « \\DESKTOP-C92JTU1\Users\prof\Desktop\partage » avec comme identifiant et mot de passe prof / 2021 :



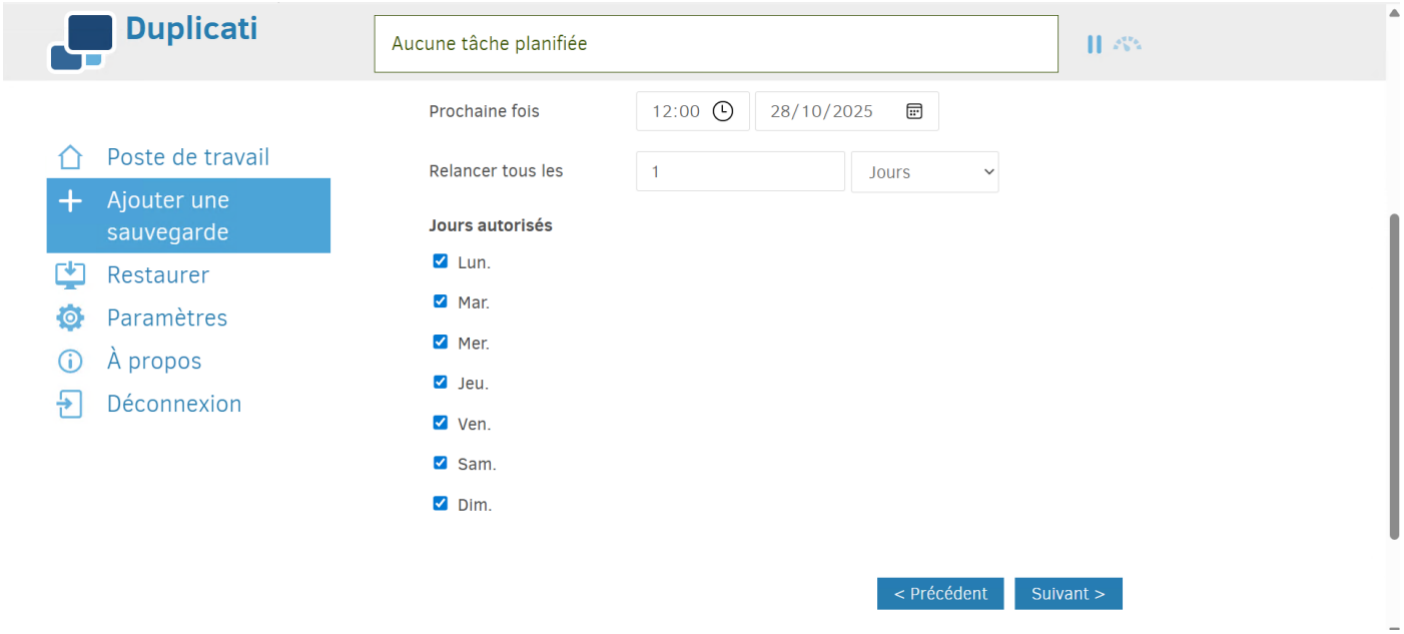
Nous pouvons cliquer sur « Tester la connexion » pour vérifier que tout fonctionne :



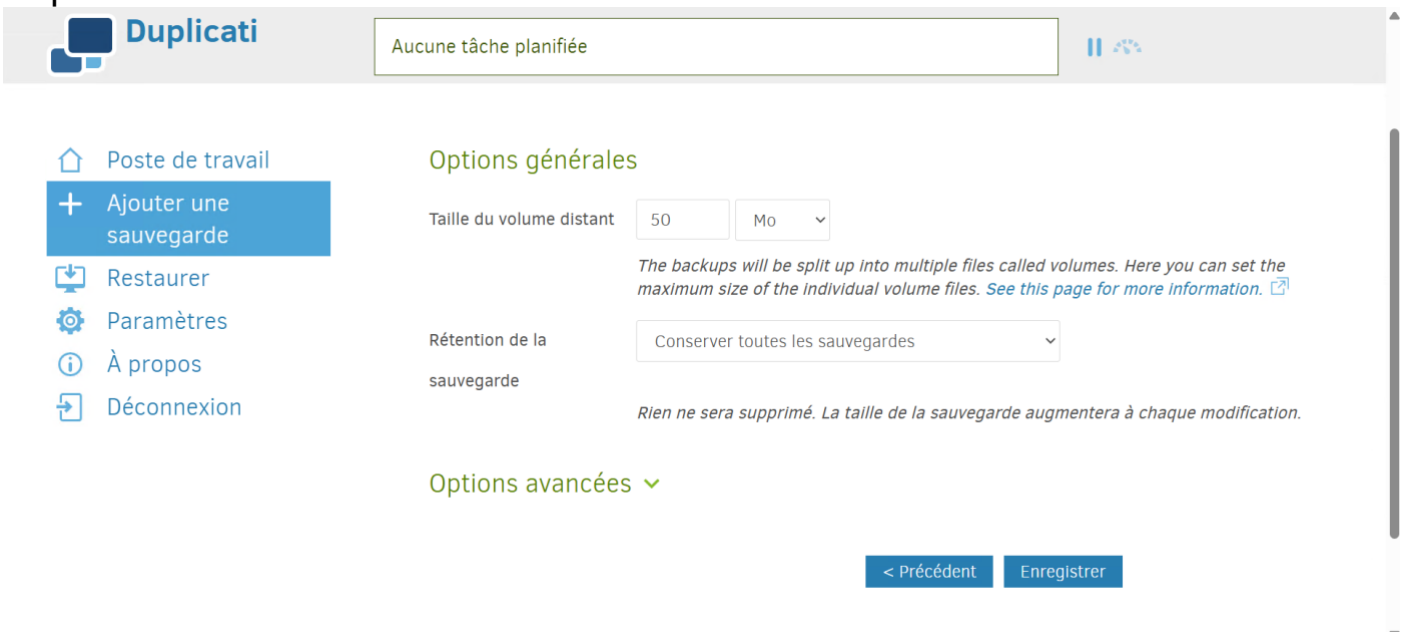
Une fois le dossier de destination défini (celui ou les données de sauvegarde seront envoyées), nous allons définir un dossier source. Dans notre cas, nous allons prendre le dossier « Mes Photos » qui contient 3 fichiers précédemment créer :



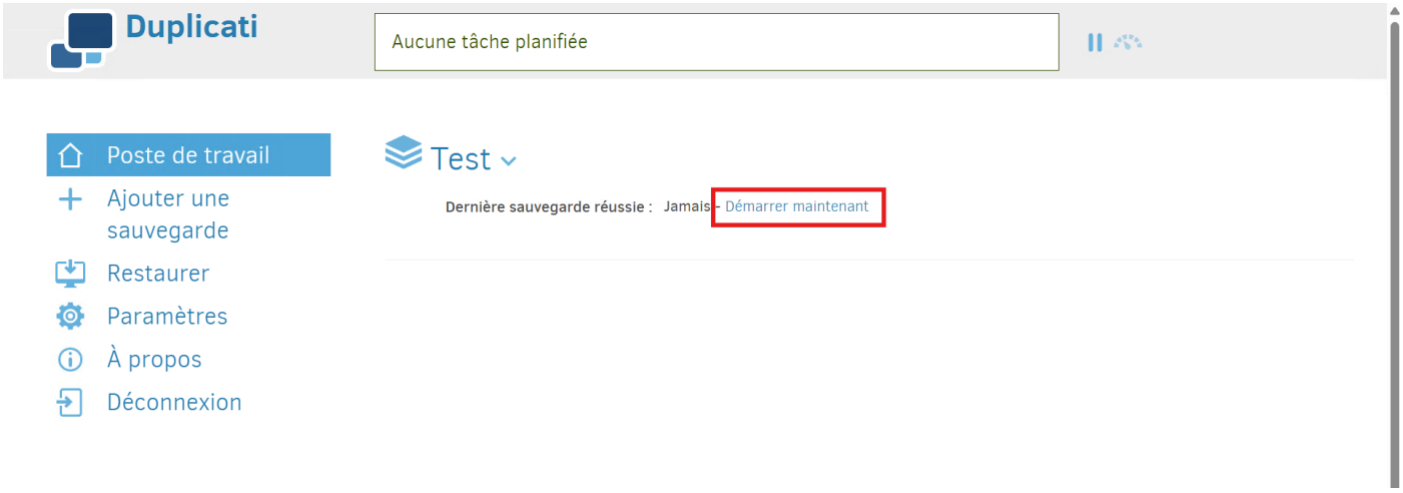
Une fois ceci défini, nous devons définir quand cette sauvegarde aura lieu (en l'occurrence, les fichiers seront sauvegardés tous les jours de la semaine à 12h) :



Maintenant, il suffit de laisser les paramètres par défaut et d'enregistrer notre configuration :



Maintenant, nous pouvons cliquer sur démarrer maintenant pour lancer la sauvegarde manuellement



Maintenant, nous pouvons vérifier le PC source (celui qui héberge le dossier partagé) si la sauvegarde est bien présente :

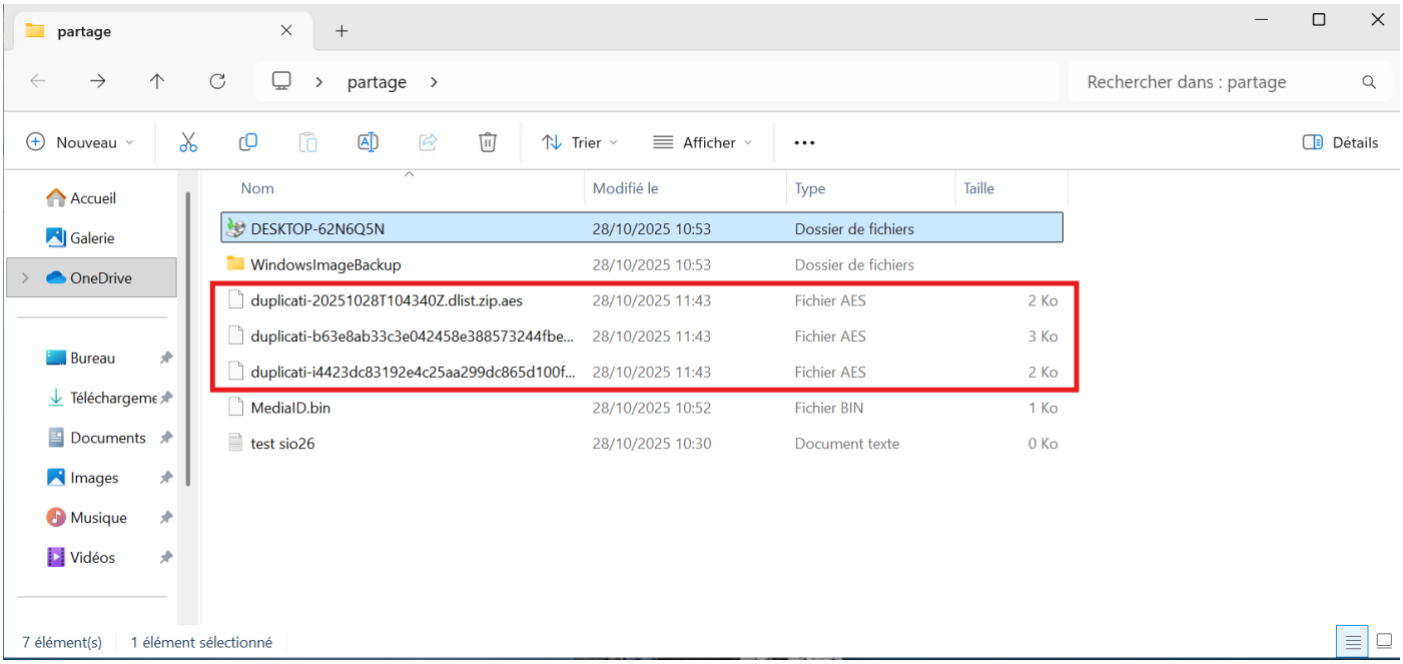
Test

Dernière sauvegarde réussie : Aujourd'hui à 11:43 (durée 00:00:02) Démarrer maintenant

Prochaine exécution programmée : Aujourd'hui à 12:00

Source : 516 octets

Sauvegarde : 4,85 KiB / 1 Version



A noté qu'il est aussi possible de restaurer cette sauvegarde à tout moment dans l'onglet « Restaurer » de Duplicati :



4) Script bat (planification)

Dans cet exemple, nous allons créer un script en .bat qui copie l'intégralité du dossier mes photos, et le déplace sur le dossier distant dans un dossier nommé : « script_bat »

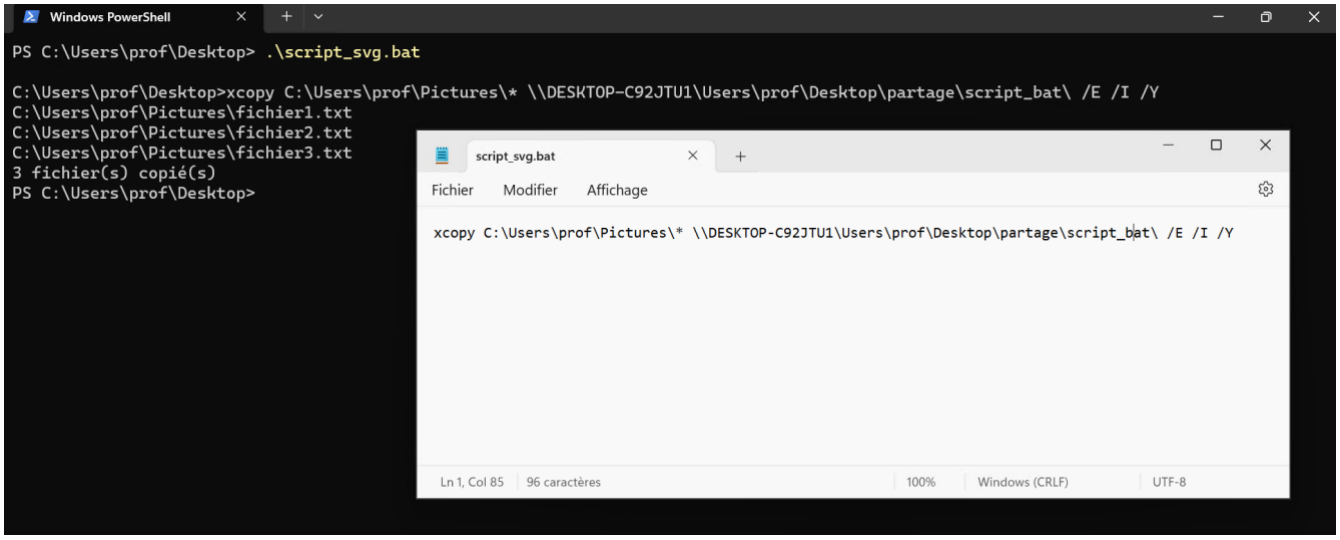
La commande utilisée dans notre script ressemble à ceci :

```
xcopy C:\Users\prof\Pictures\* \\DESKTOP-C92JTU1\Users\prof\Desktop\partage\script_bat\ /E /I /Y
```

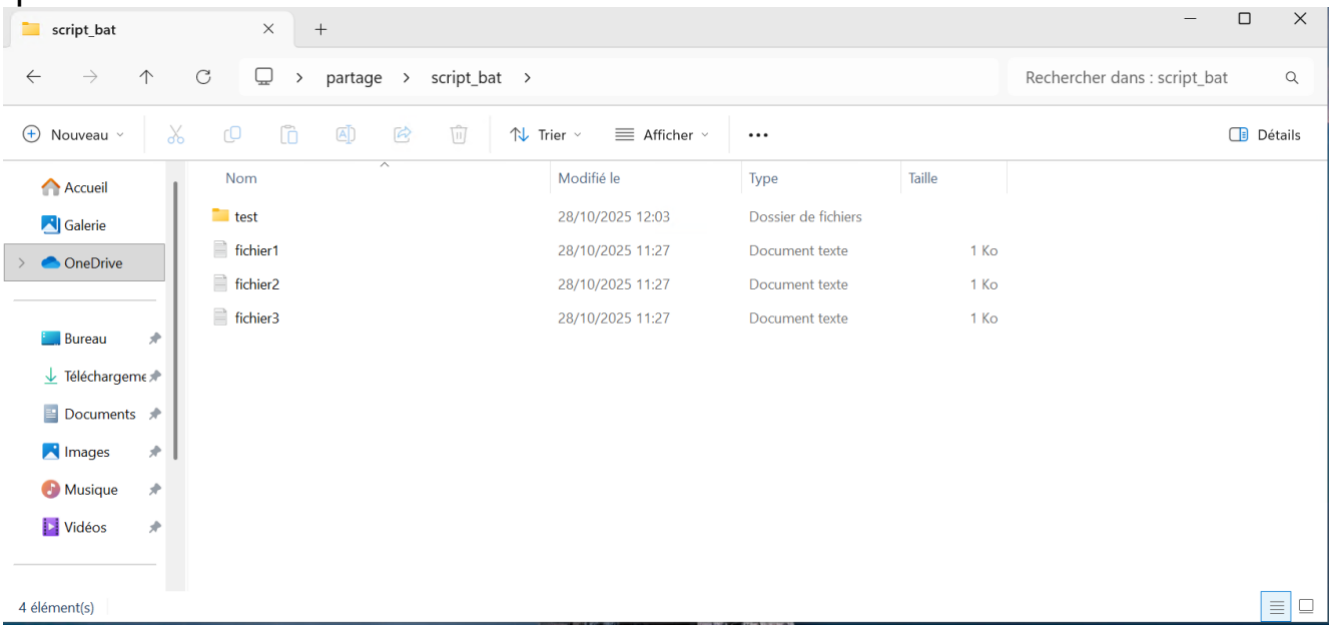
Arguments utilisés dans la commande :

```
/E Copie tous les sous-dossiers, y compris les dossiers vides
/I Suppose que la destination est un dossier lorsqu'on copie plusieurs fichiers
/Y Supprime les demandes de confirmation avant d'écraser des fichiers existants
```

1er test sans planifier une tâche :



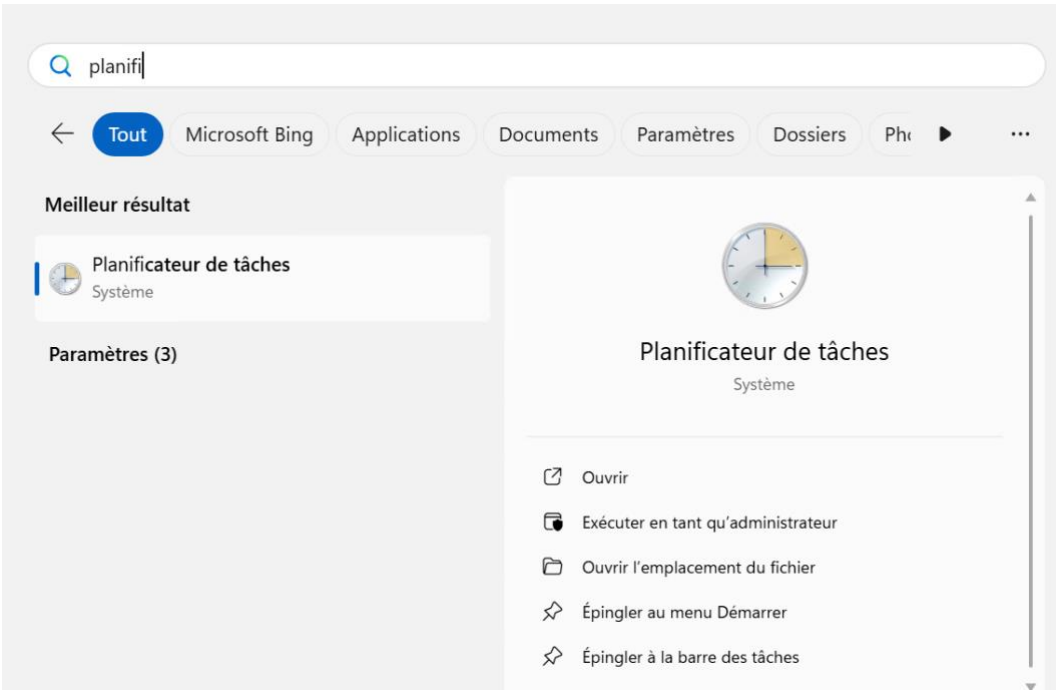
Une fois le script démarré, sur le PC source, nous avons bien les 3 fichiers ainsi que le dossier « test » qui était présent dans le dossier mes photos :

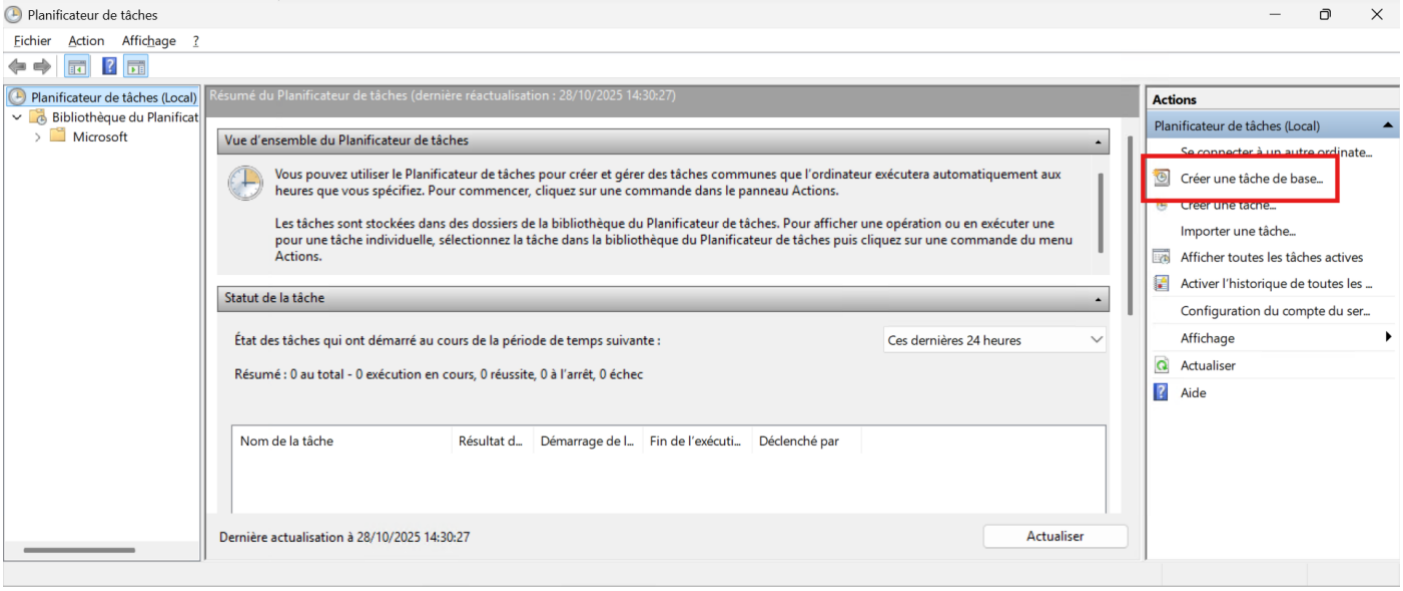


2ème test avec le planificateur de tâche :

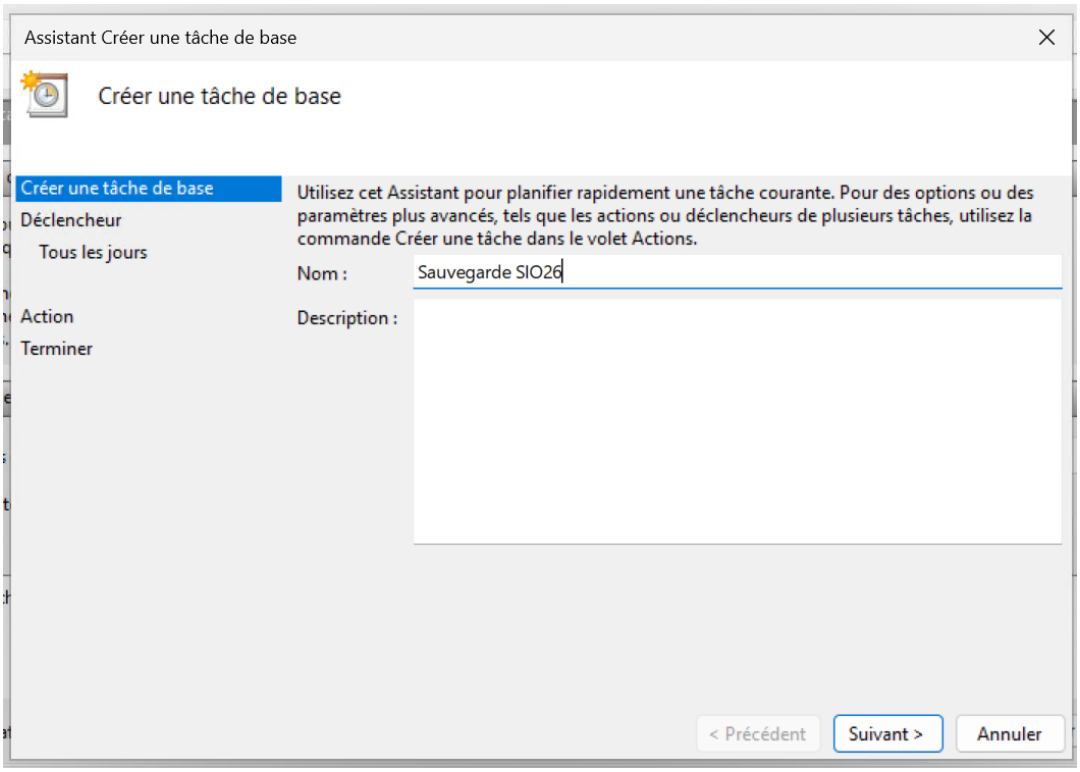
Maintenant que nous savons que ce script fonctionne, nous pouvons l'intégrer dans le planificateur des tâches pour que les sauvegardes se réalisent automatiquement à un moment donné.

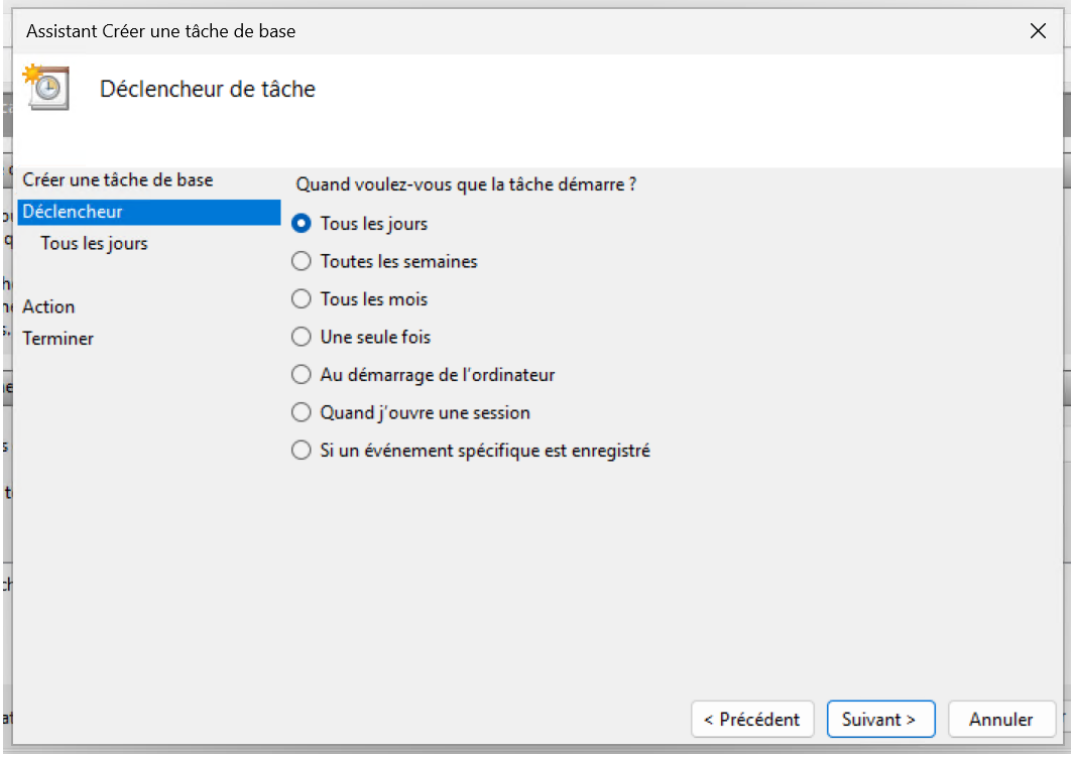
Pour commencer, nous allons ouvrir le planificateur de tâches et puis nous allons créer une nouvelle tâche :

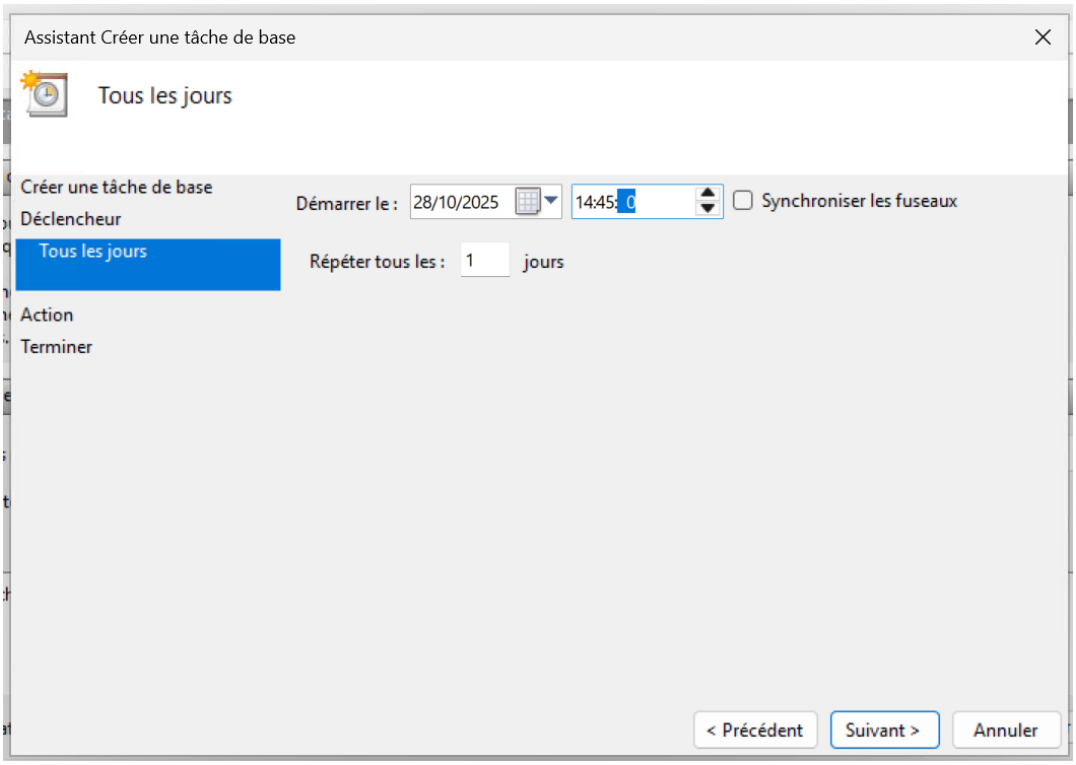


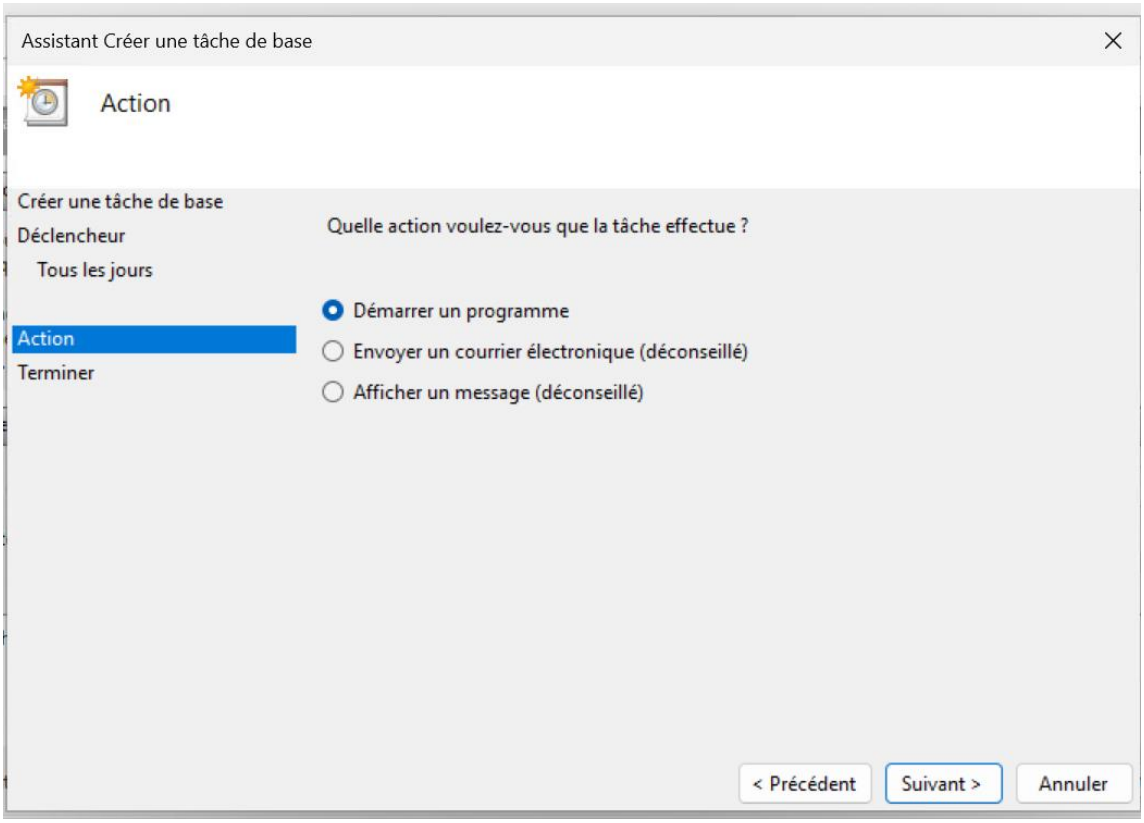


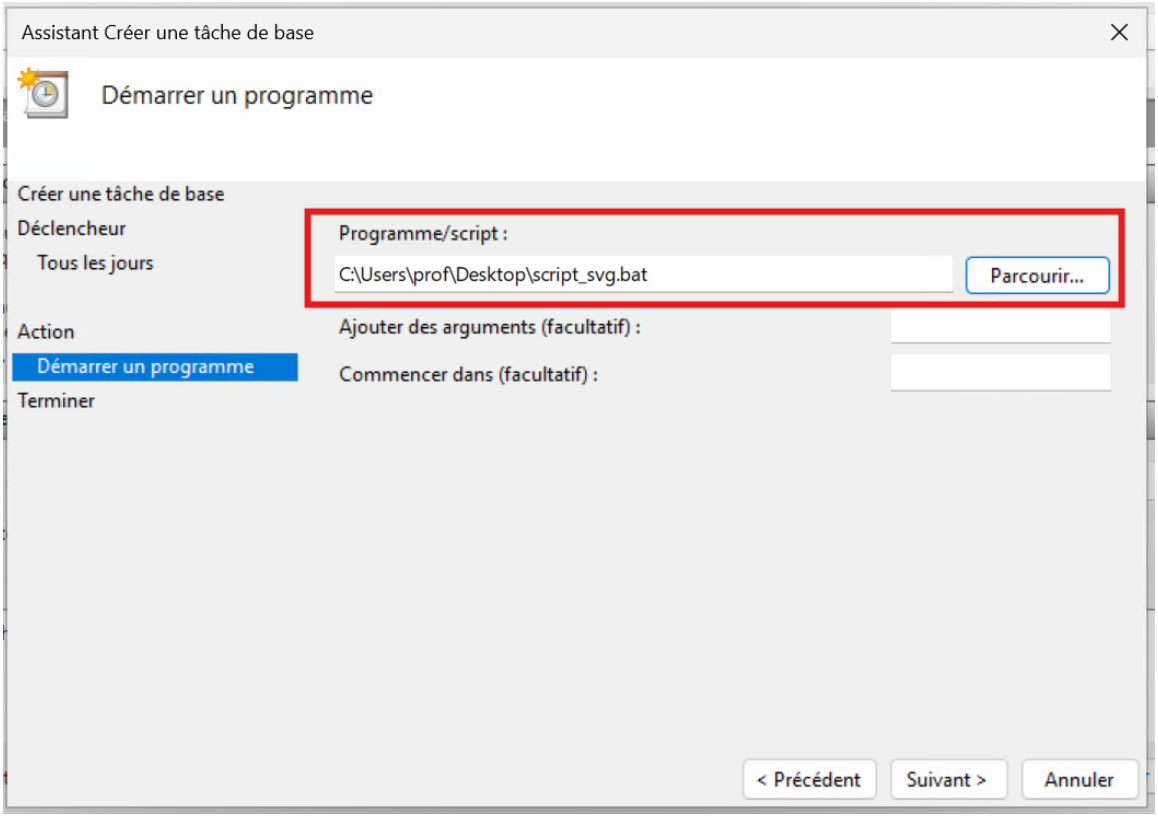
Maintenant, nous allons créer une tâche de base, nous allons devoir définir son nom, le déclencheur (en l'occurrence 1 fois par jour), et l'action qu'elle devra réaliser (en l'occurrence, exécuter le script que nous avons créé) :

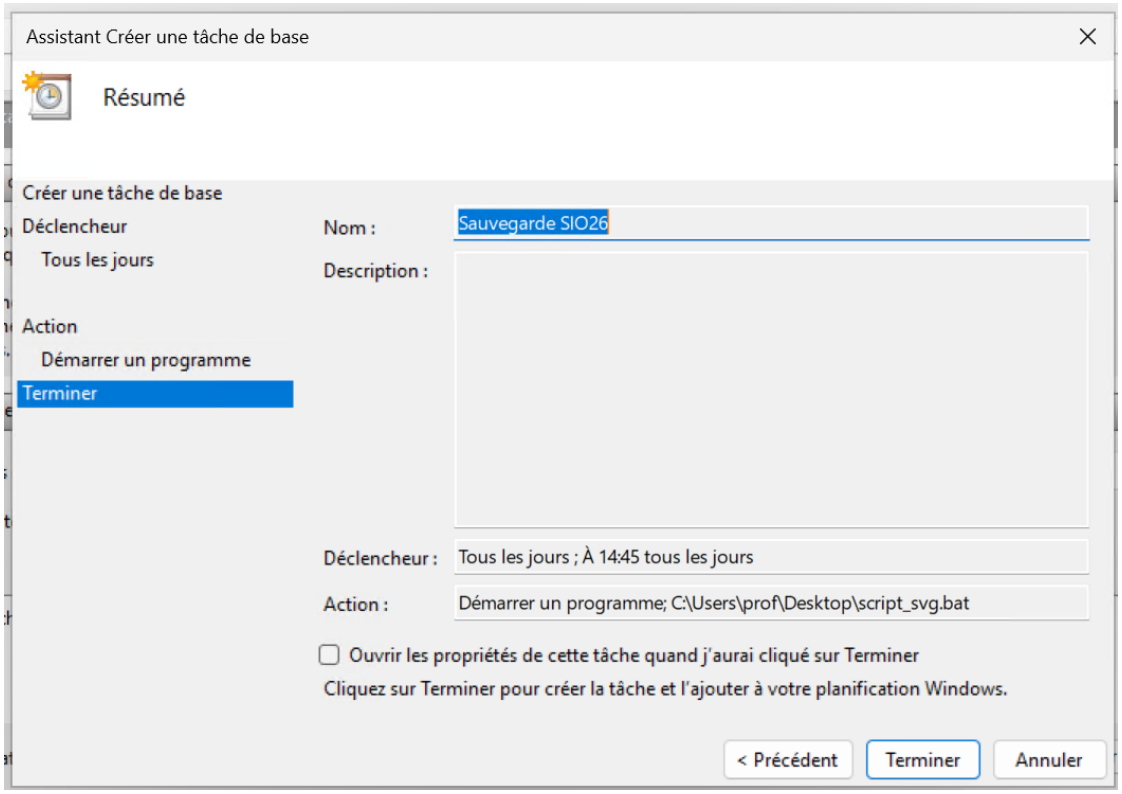



Assistant Créer une tâche de base
Déclencheur de tâche
Créer une tâche de base
Déclencheur
Tous les jours
Action
Terminer
Quand voulez-vous que la tâche démarre ?
Tous les jours
Toutes les semaines
Tous les mois
Une seule fois
Au démarrage de l'ordinateur
Quand j'ouvre une session
Si un événement spécifique est enregistré
< Précédent
Suivant >
Annuler


Assistant Créer une tâche de base
Tous les jours
Créer une tâche de base
Déclencheur
Tous les jours
Action
Terminer
Démarrer le : 28/10/2025 14:45:0
Synchroniser les fuseaux
Répéter tous les : 1 jours
< Précédent
Suivant >
Annuler


Assistant Créer une tâche de base
Action
Créer une tâche de base
Déclencheur
Tous les jours
Action
Terminer
Quelle action voulez-vous que la tâche effectue ?
Démarrer un programme
Envoyer un courrier électronique (déconseillé)
Afficher un message (déconseillé)
< Précédent
Suivant >
Annuler


Assistant Créer une tâche de base
Démarrer un programme
Créer une tâche de base
Déclencheur
Tous les jours
Action
Démarrer un programme
Terminer
Programme/script :
C:\Users\prof\Desktop\script_svg.bat
Parcourir...
Ajouter des arguments (facultatif) :
Commencer dans (facultatif) :
< Précédent
Suivant >
Annuler



Une fois la tâche créer, il suffit d'attendre 14h45, ou dans notre cas, nous allons lancer la tâche manuellement pour vérifier que cette dernière fonctionne :

Vidéo : https://drive.proton.me/urls/NMM5Y378YW#QtFUnCJSlrX3

II/ Linux

1) Partage NFS

NFSv4 : Network File System, libre (contre SMB de Windows), port 2049

VM Ubuntu sur réseau 407.

Est-ce que NFS est installé ? Sinon : sudo apt install nfs-server

```
hvusr@hv2:~$ sudo apt install nfs-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Note : sélection de « nfs-kernel-server » au lieu de « nfs-server »
Le paquet suivant a été installé automatiquement et n'est plus nécessaire :
  libllvm19
Veuillez utiliser « sudo apt autoremove » pour le supprimer.
Les paquets supplémentaires suivants seront installés :
  keyutils libevent-core-2.1-7t64 nfs-common rpcbind
Paquets suggérés :
  open-iscsi watchdog
Les NOUVEAUX paquets suivants seront installés :
  keyutils libevent-core-2.1-7t64 nfs-common nfs-kernel-server rpcbind
0 mis à jour, 5 nouvellement installés, 0 à enlever et 10 non mis à jour.
Il est nécessaire de prendre 612 ko dans les archives.
Après cette opération, 2 103 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] O
```

Créer le dossier partagé /home/partage

```
hvusr@hv2:~$ sudo mkdir /home/partage
hvusr@hv2:~$
```

Partager ce dossier dans /etc/exports pour votre voisin :

1. /home/partage ip_client (rw,no_root_squash)

Modification du fichier avec l'éditeur de texte nano (sudo nano /etc/exports) :

```
 GNU nano 7.2                                /etc/exports
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#
# Example for NFSv4:
# /srv/nfs4        gss/krb5i(rw,sync,fsid=0,crossmnt,no_subtree_check)
# /srv/nfs4/homes  gss/krb5i(rw,sync,no_subtree_check)
#
/home/partage 192.168.50.1(rw,no_root_squash)
```

Démarrer le serveur NFS : sudo systemctl start nfs-server

```
hvusr@hv2:~$ sudo systemctl restart nfs-server
hvusr@hv2:~$
```

Accéder au dossier partagé de votre voisin :

2. Créer le dossier montage local : /mnt/partage

Création du dossier /mnt/partage avec mkdir sur le PC de destination :

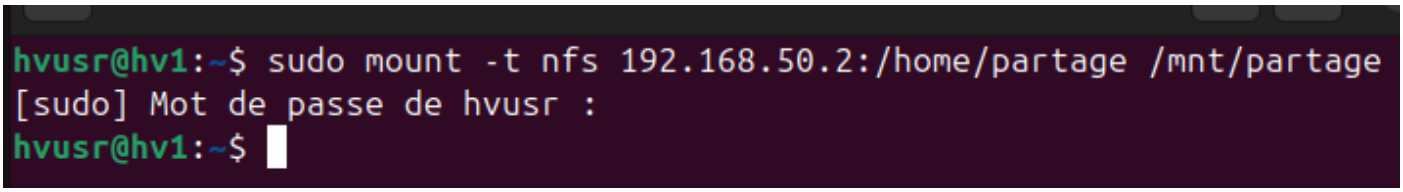
```
hvusr@hv1:~$ sudo mkdir /mnt/partage
[sudo] Mot de passe de hvusr :
```

3. Monter le dossier partagé de votre voisin :

Avant de monter notre serveur NFS, nous devons installer le client NFS avec la commande : sudo apt install nfs-common

```
hvusr@hv1:~$ sudo apt install nfs-common
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Le paquet suivant a été installé automatiquement et n'est plus nécessaire :
  libllvm19
Veuillez utiliser « sudo apt autoremove » pour le supprimer.
Les paquets supplémentaires suivants seront installés :
  keyutils libevent-core-2.1-7t64 rpcbind
Paquets suggérés :
  open-iscsi watchdog
Les NOUVEAUX paquets suivants seront installés :
  keyutils libevent-core-2.1-7t64 nfs-common rpcbind
0 mis à jour, 4 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 443 ko dans les archives.
Après cette opération, 1 497 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

Maintenant, nous pouvons exécuter la commande suivante sur le client pour monter le partage NFS : mount -t nfs ip_server:/home/partage /mnt/partage



Une fois le partage monté, nous pouvons essayer de créer un fichier dans /mnt/partage sur le client, et vérifier que les changements sont bien reflétés sur le serveur :

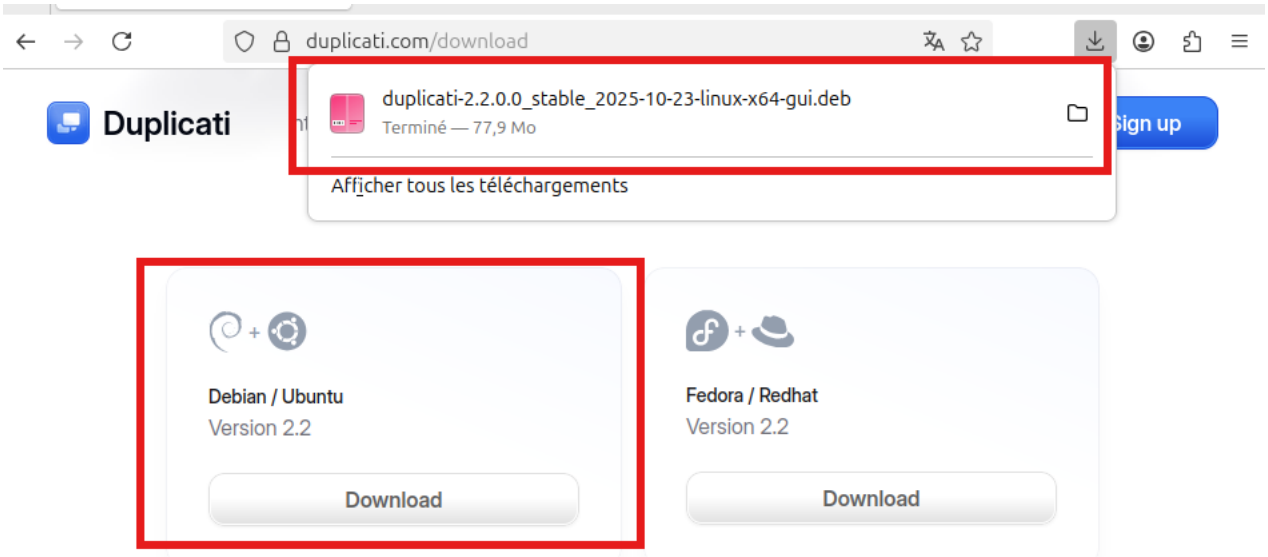




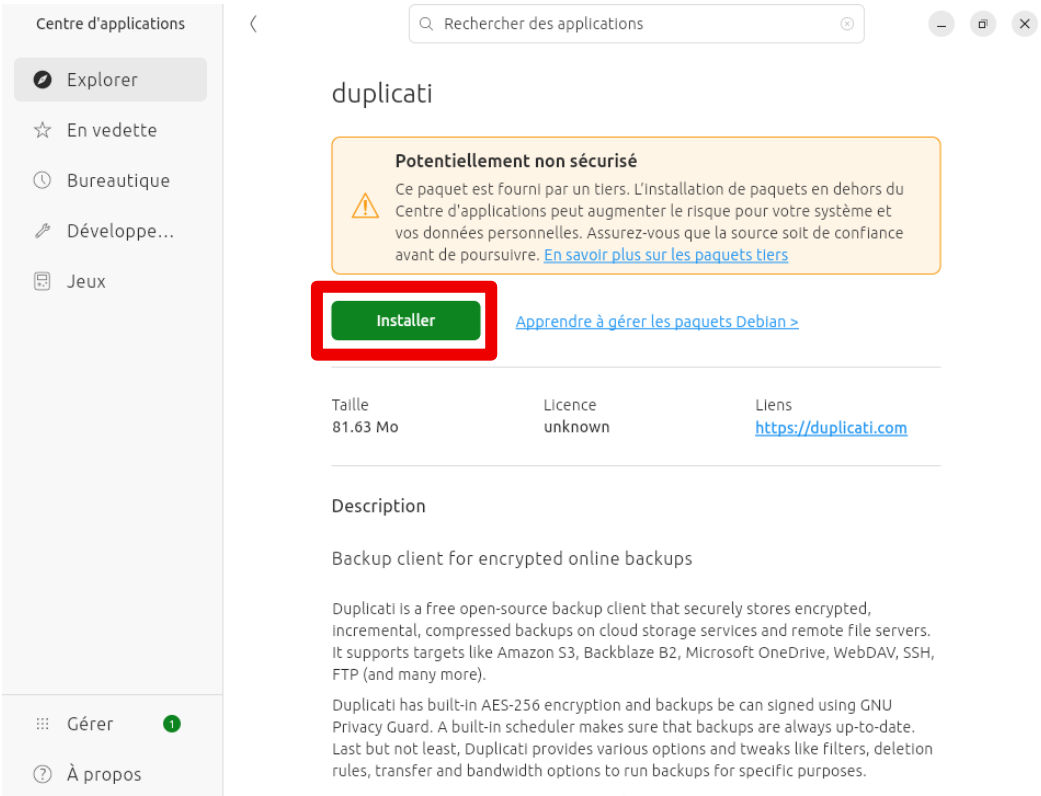
2) Utiliser Duplicati

1. Première étape (Installation de Duplicati) :

Pour installer Duplicati sur le client NFS, il suffit de se rendre sur le site officiel et de télécharger la version pour Ubuntu / Debian

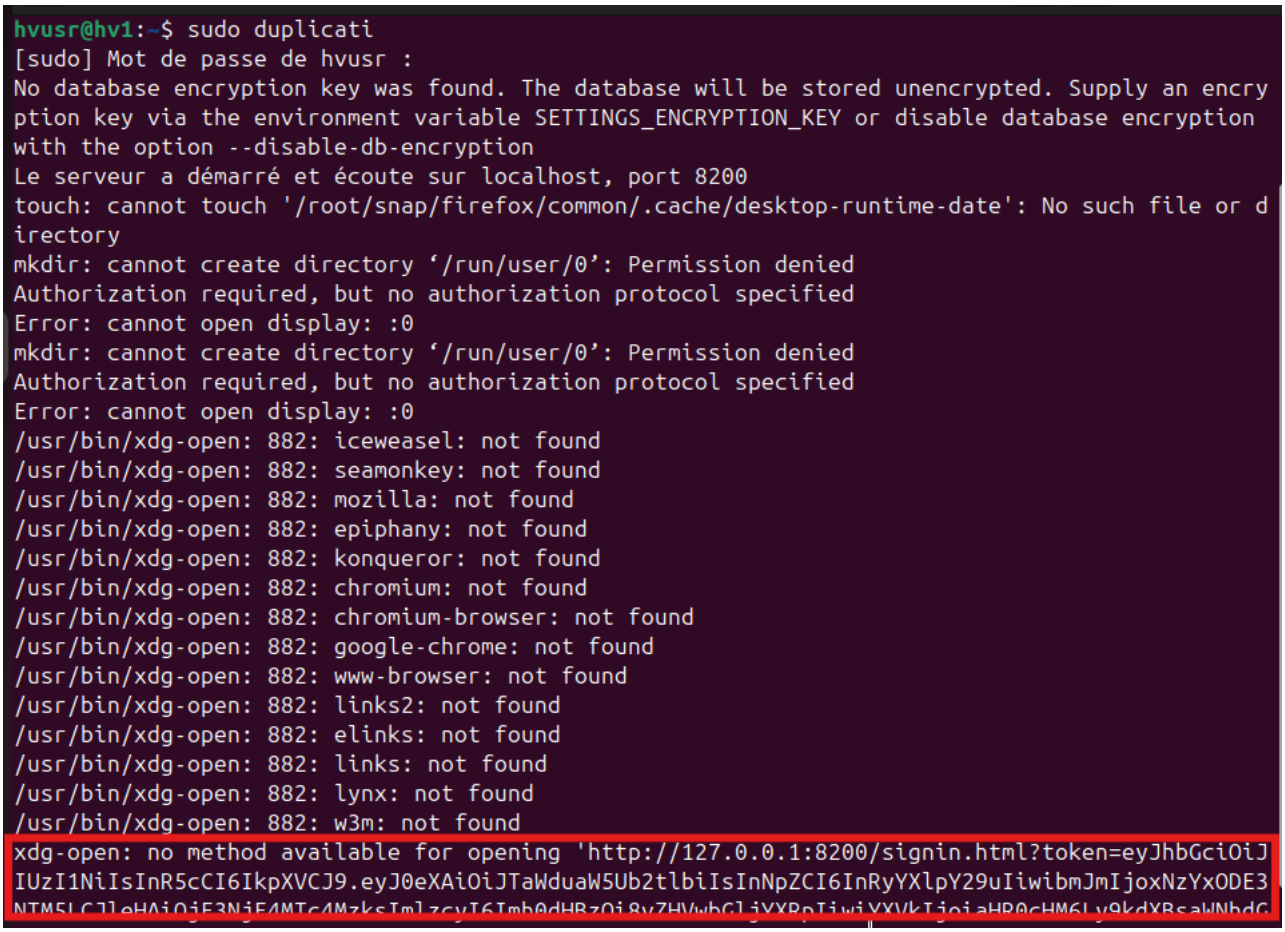


Une fois le fichier téléchargé, nous allons l'installer (via l'interface graphique) en cliquant dessus, et nous allons cliquer sur le bouton « Installer » pour installer ce dernier :

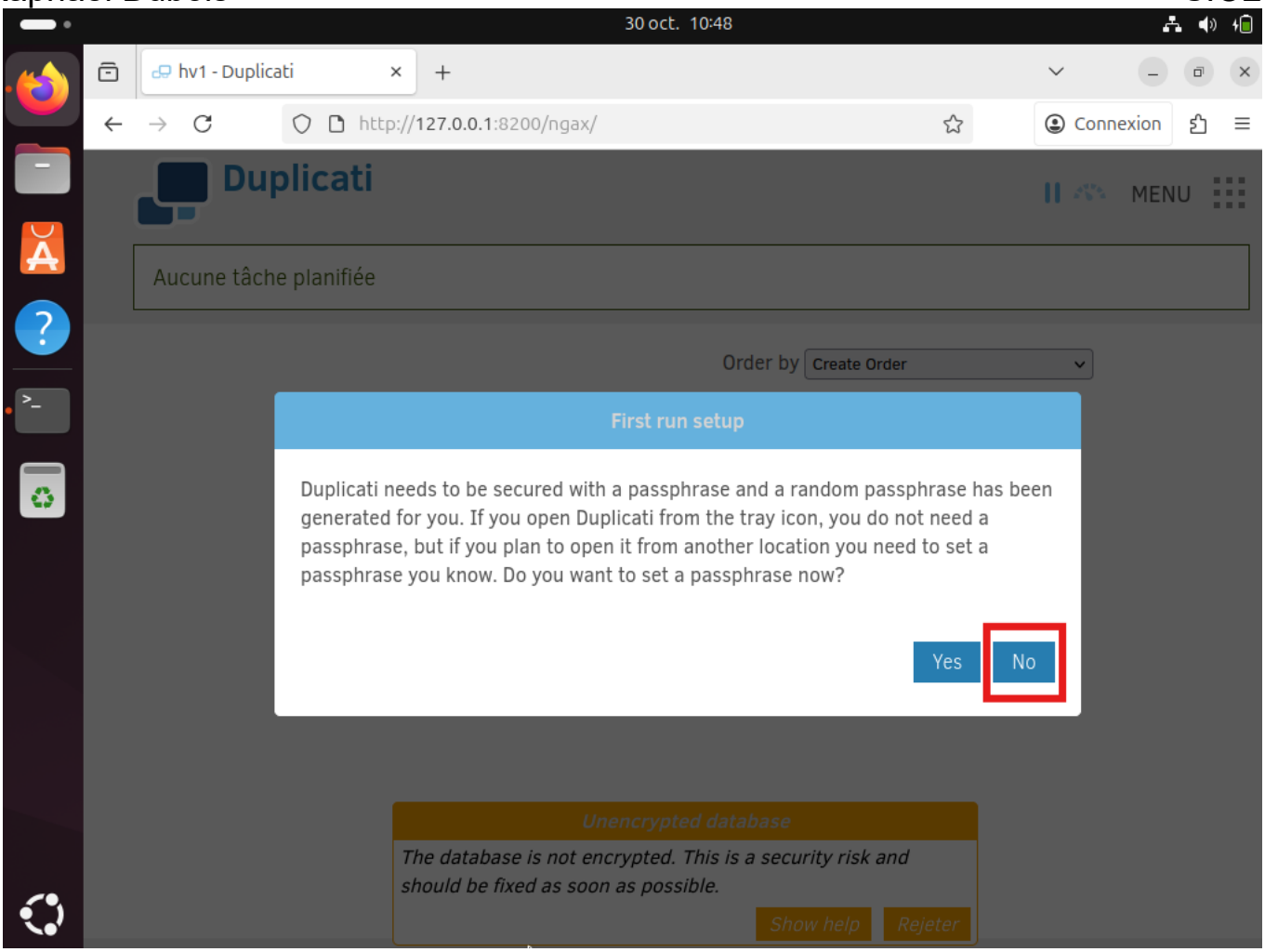


Une fois duplicati installé, nous allons le lancer dans un terminal en sudo (afin que ce dernier puisse avoir accès à tous les fichiers et dossiers, dont notre dossier

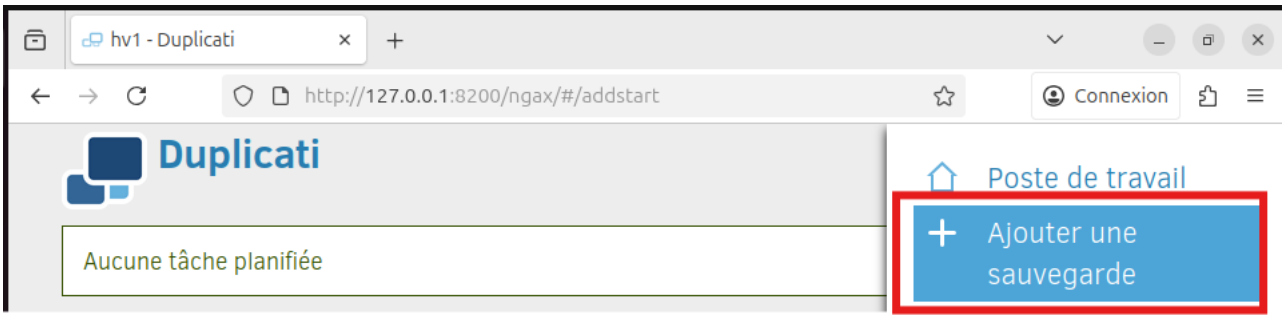
/mnt/partage), pour cela, nous allons utiliser la commande **sudo duplicati** pour lancer duplicati en mode root, puis nous allons cliquer sur lien présent dans le terminal :

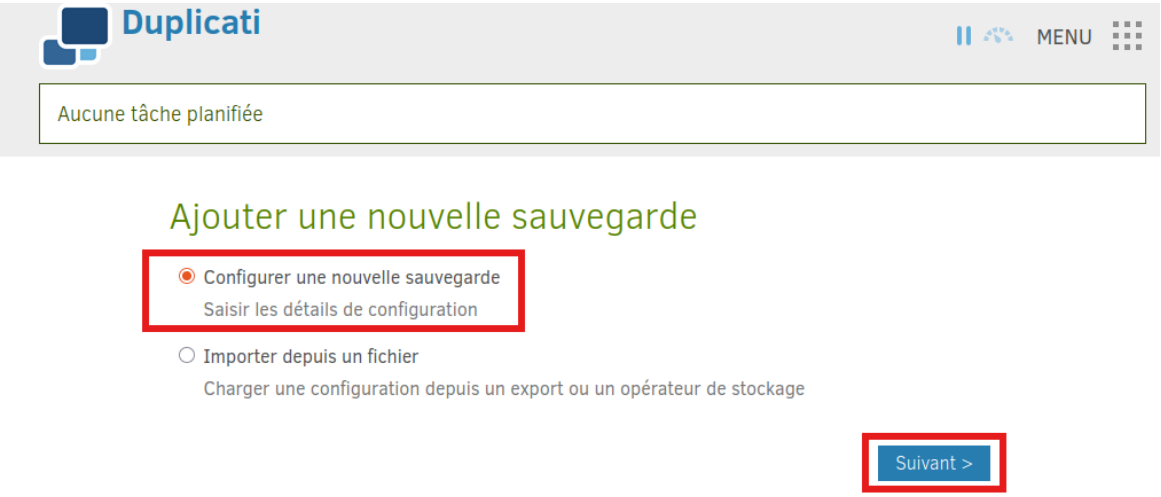


Une fois que nous avons cliqué sur le lien, duplicati s’ouvre exactement comme dans Windows :

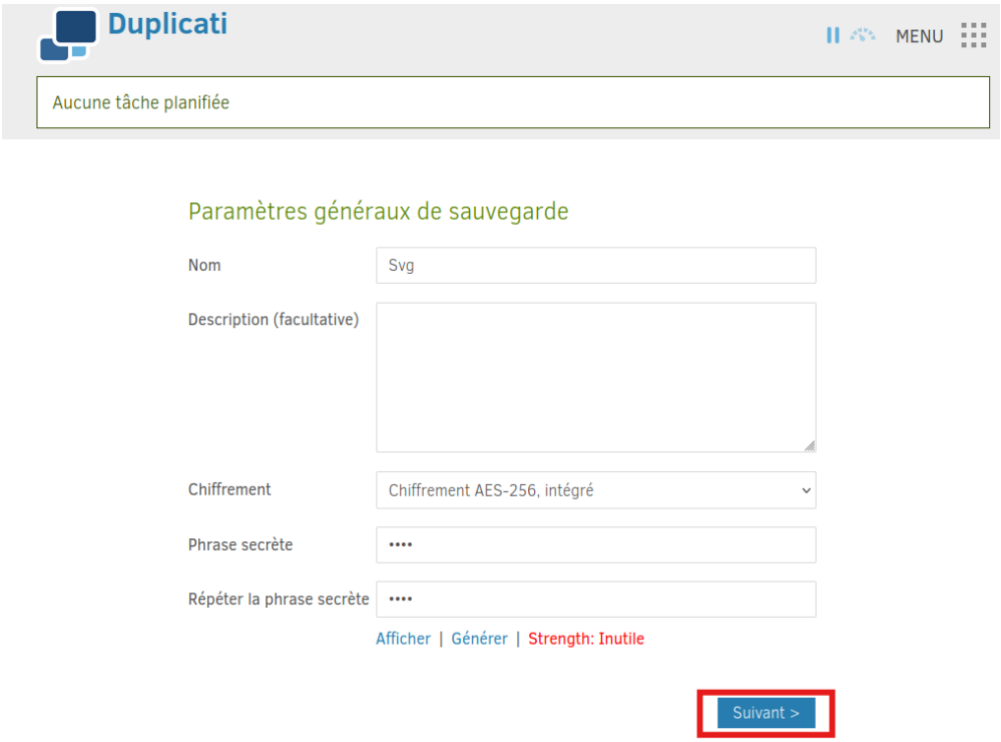


Comme sur Windows, nous allons créer une nouvelle sauvegarde :

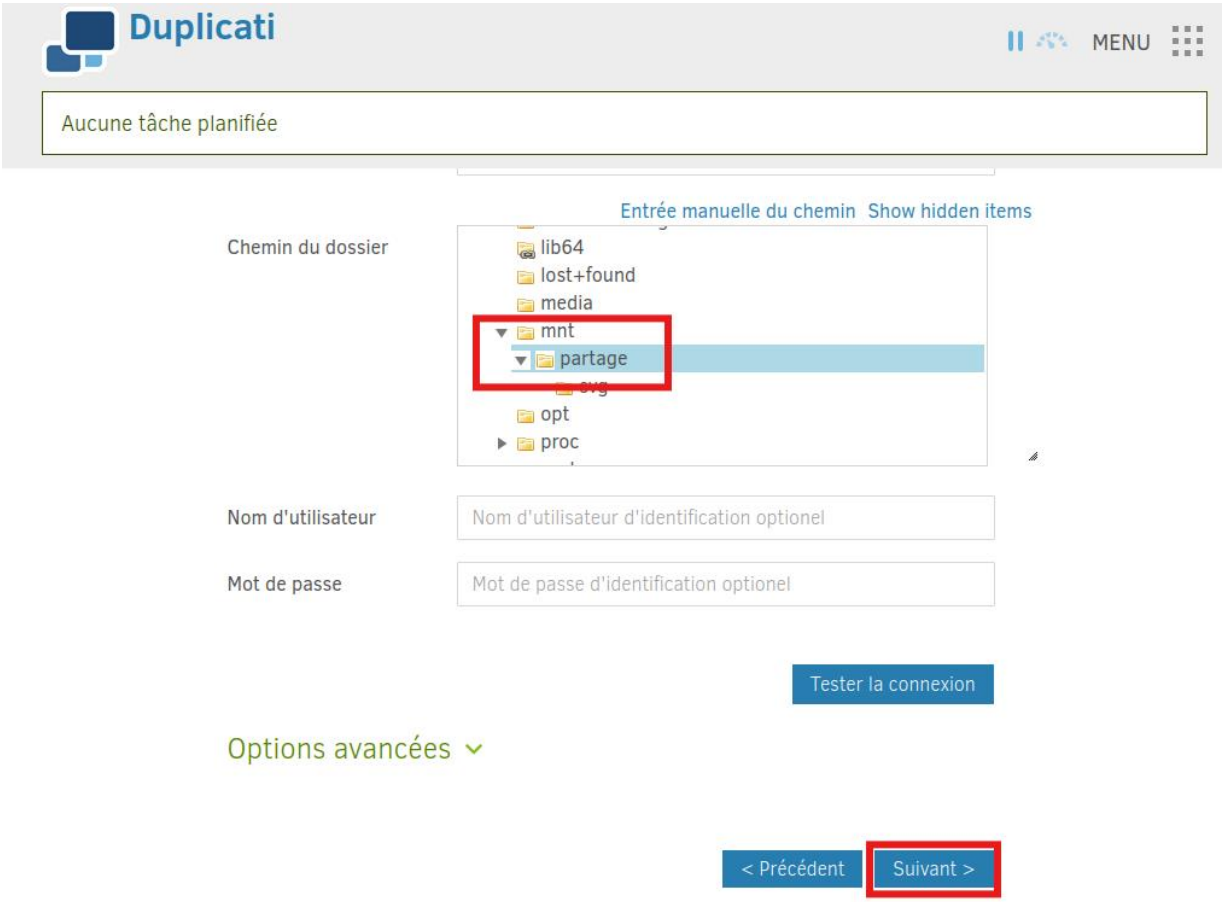




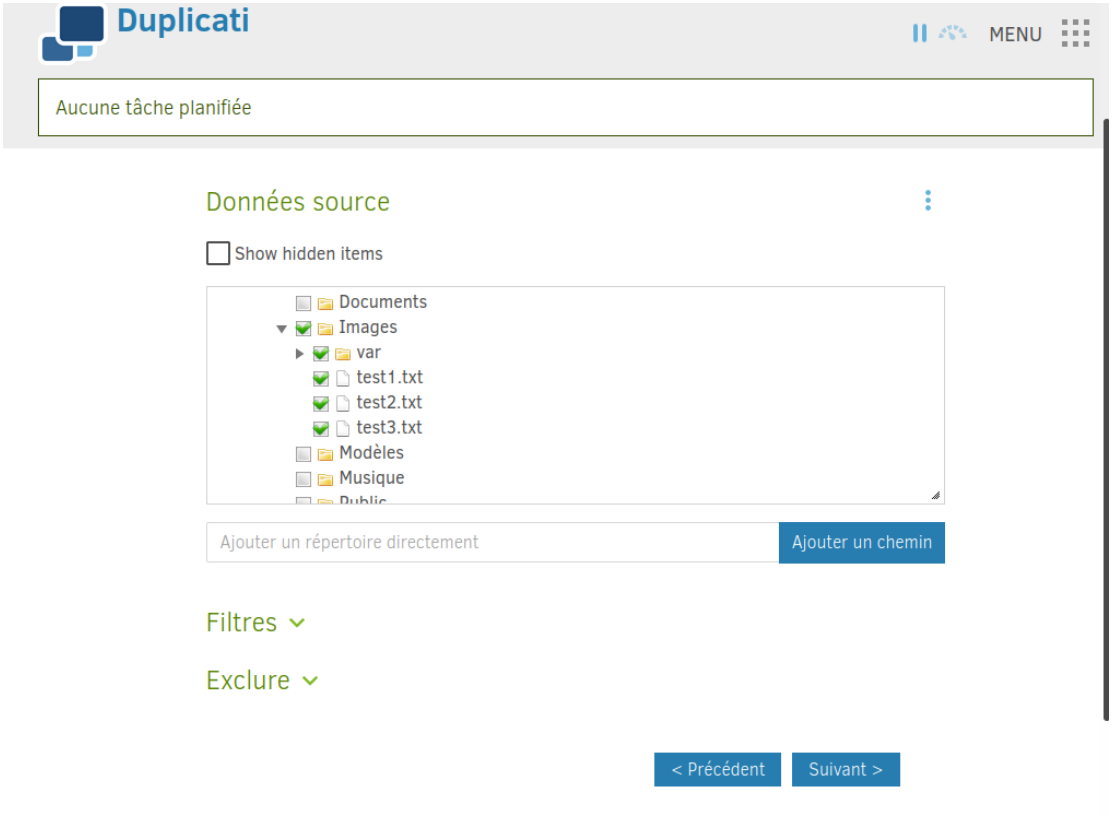
Comme sur Windows, nous allons définir un nom à notre sauvegarde,



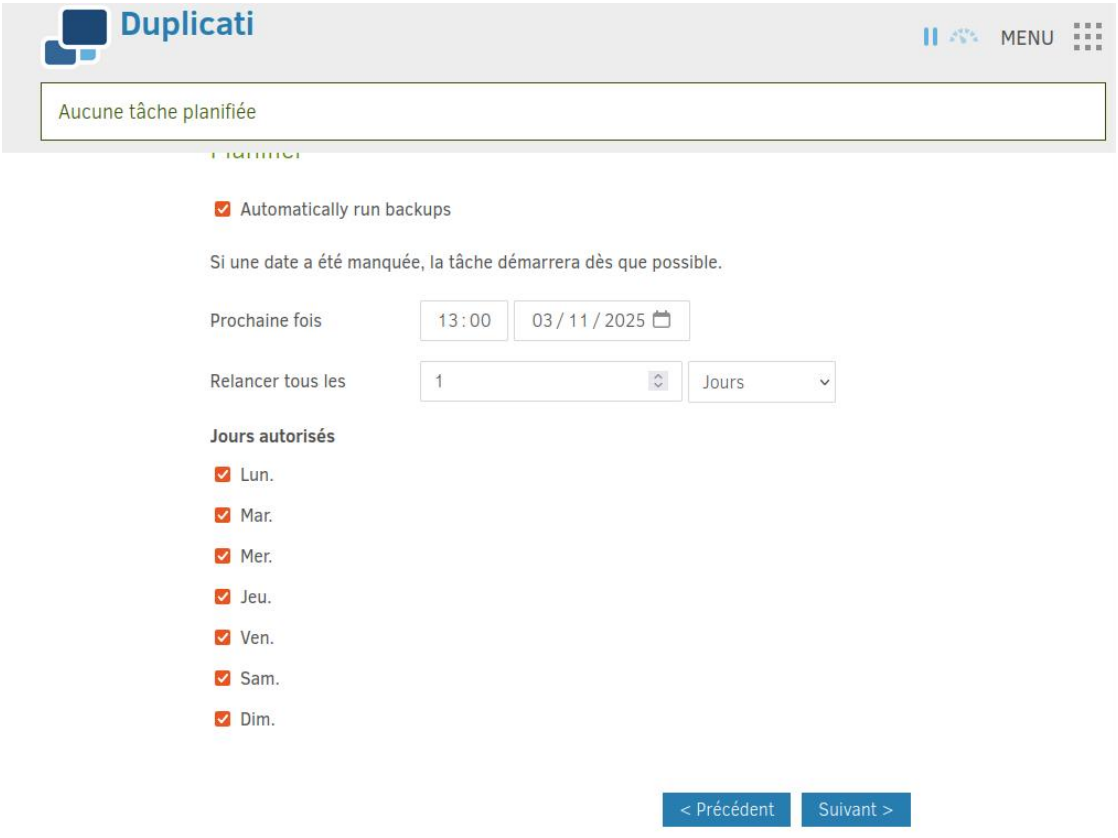
Comme sur Windows, nous allons sélectionner le dossier de sauvegarde (en l'occurrence, /mnt/partage)



Maintenant, nous allons sélectionner les fichiers à sauvegarder (en l'occurrence, tout le dossier images) :



Ensuite, nous sélectionnons quand es que la sauvegarde aura lieu :

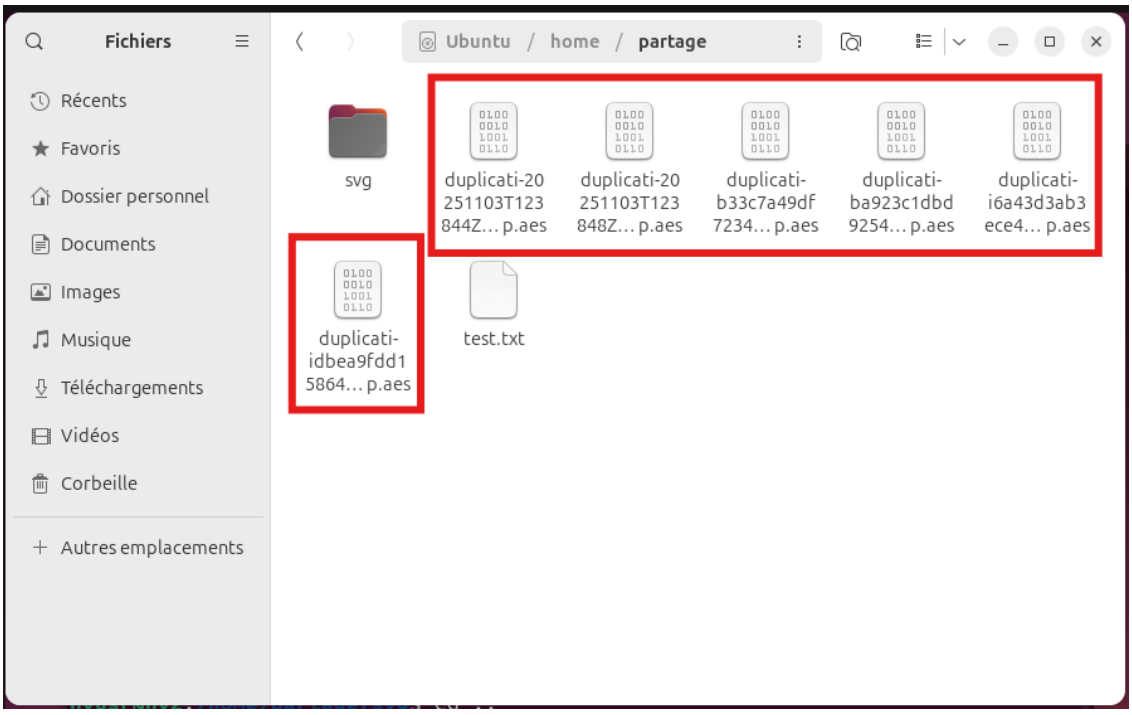


Puis, nous pouvons enregistrer :

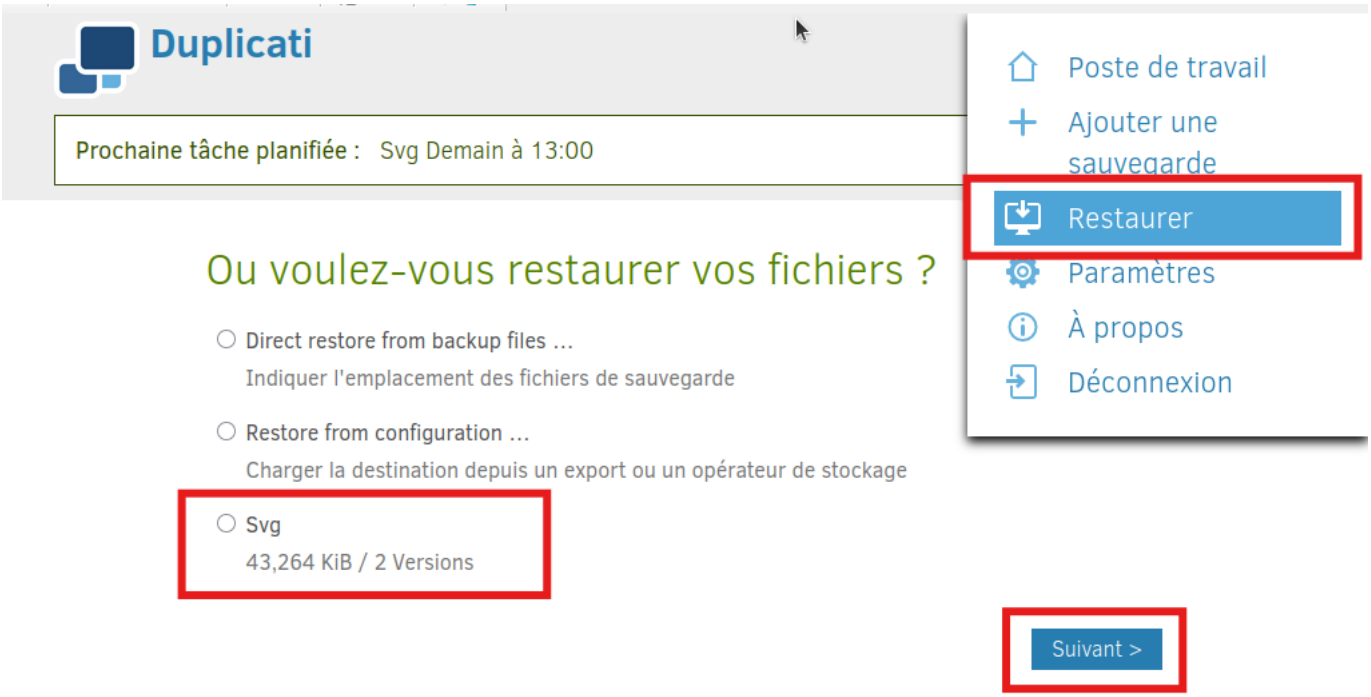


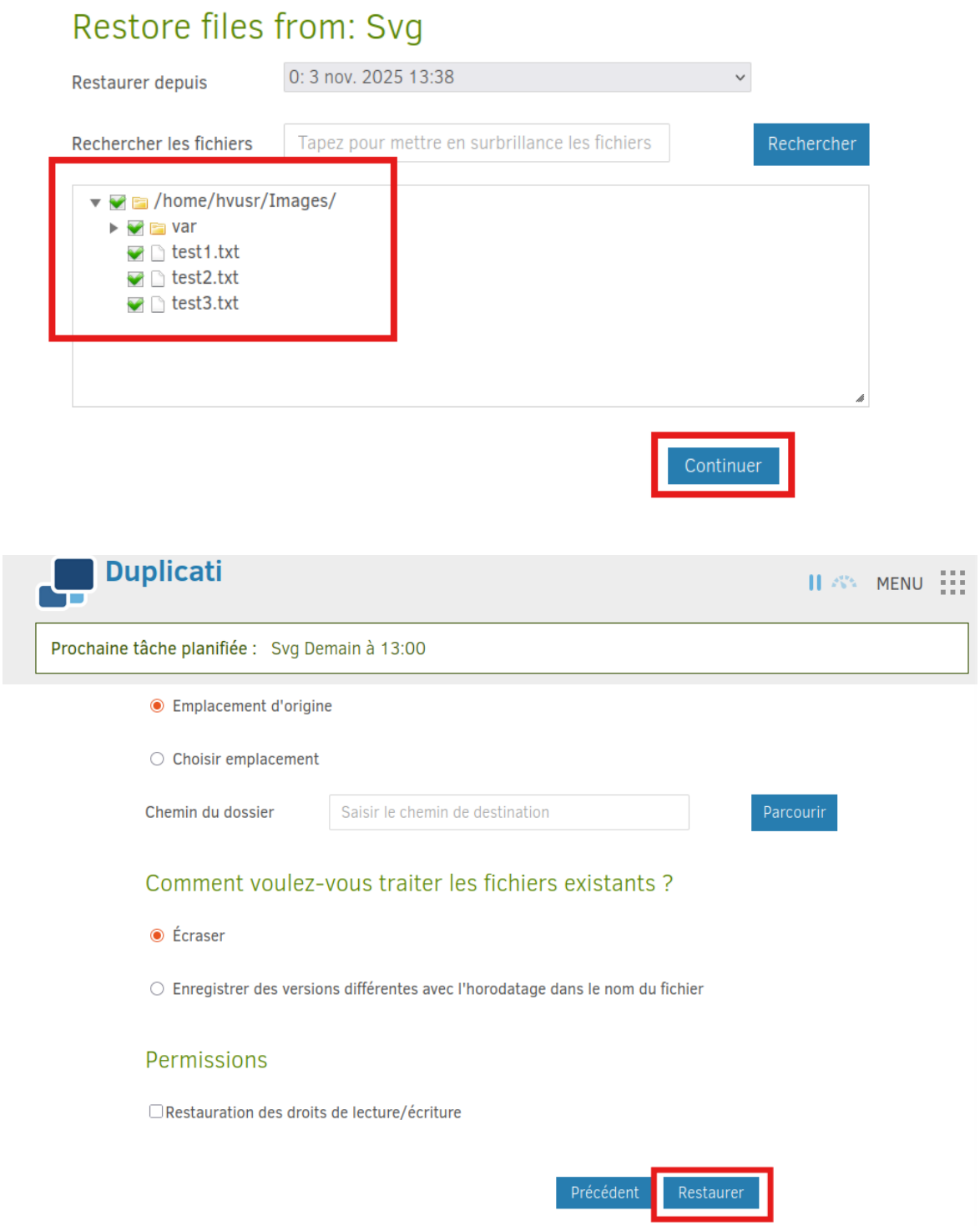
Une fois que tout a été effectué, on peut vérifier sur le serveur NFS si la sauvegarde a bien été effectuée :





Une fois ceci effectué, nous allons essayer de restaurer la sauvegarde sur notre client NFS :





Une fois ceci fait, il suffit de patienter pour que la sauvegarde soit restaurée !

3) Sauvegarde manuelle (cp, tar, at, crontab)

1. Commande cp

Nous pouvons utiliser la commande suivante pour copier tout le contenu du dossier Images, et le coller dans notre dossier partagé

```
sudo cp -r /home/hvusr/Images/* /mnt/partage/svg/
```

```
hvusr@hv1:~/Images$ sudo cp -r /home/hvusr/Images/* /mnt/partage/svg/
hvusr@hv1:~/Images$ ls /mnt/partage/svg/
test1.txt  test2.txt  test3.txt  var
hvusr@hv1:~/Images$ ls /home/hvusr/Images/
test1.txt  test2.txt  test3.txt  var
hvusr@hv1:~/Images$
```

2. Commande tar

Pour archiver, compresser et envoyer notre fichier de sauvegarde vers le serveur NFS, nous pouvons utiliser la commande suivante :

```
sudo tar -czvf /mnt/partage/svg.tar.gz /home/hvusr/Images/
```

- `-c` : crée une nouvelle archive
- `-z` : Compresse l'archive avec **gzip** (format compressé)
- `-v` : mode verbeux (affiche les fichiers traités, optionnelle)
- `-f` : indique le nom du fichier de sortie (ici /mnt/partage/svg.tar.gz)

Une fois la commande exécutée, nous obtenons un fichier « svg.tar.gz » sur notre serveur, ce dernier contient tout le contenu du dossier Images de notre client.

3. Planification avec at

Nous allons utiliser la commande pour planifier une sauvegarde à 10h05, pour cela nous allons reprendre notre commande cp en y ajoutant un « pipe » suivie de l'heure ou la commande va s'exécuter (en l'occurrence 10h05) :

```
root@debian:/mnt/partage# cp -r /root/Documents/* /mnt/partage/ | at 10:05
warning: commands will be executed using /bin/sh
job 3 at Fri Nov 28 10:05:00 2025
root@debian:/mnt/partage#
```

Une fois 10h05 passée, on remarque que les fichiers présents ont bien été sauvegardées sur le serveur NFS :

```
root@debian:/mnt/partage# ls
123.txt
root@debian:/mnt/partage#
```

4. Planification avec Crontab

Nous allons faire en sorte que la sauvegarde se réalise tous les jours à 10h45.

Premièrement, nous allons ouvrir l'éditeur crontab avec la commande crontab -e, puis nous allons insérer les informations suivantes dans le fichier :

```
# m h  dom mon dow   command
45 10 * * * cp -r /root/Documents/* /mnt/partage/
```

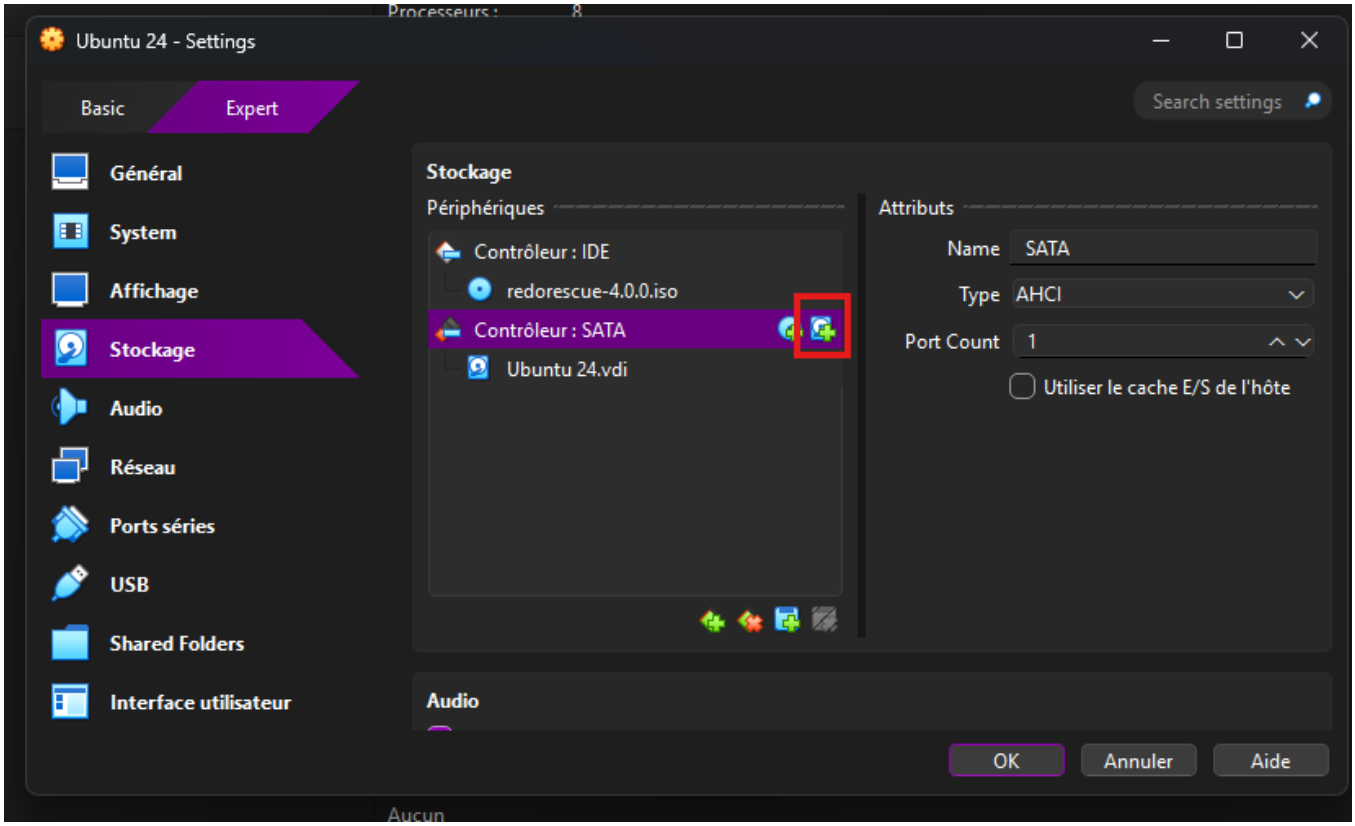
Ces informations vont faire exécuter la commande cp tous les jours à 10h45. Une fois les informations insérées, on peut sauvegarder et quitter le fichier.

Vérification après 10h45, le fichier a bien été copié sur le serveur NFS :

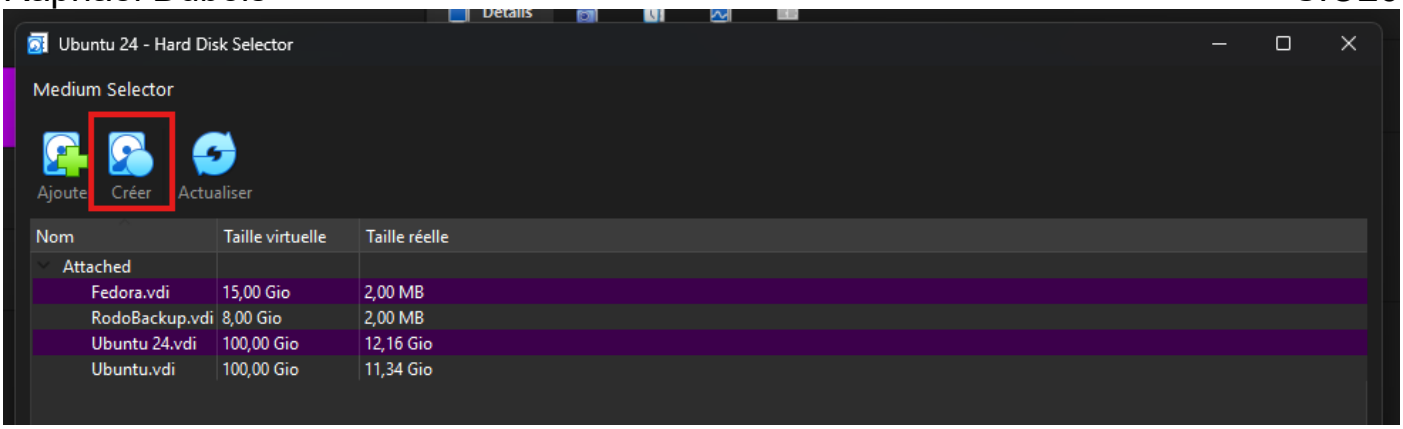
```
root@debian:~/Documents# ls /mnt/partage/
root@debian:~/Documents# ls /mnt/partage/
123.txt
root@debian:~/Documents#
```

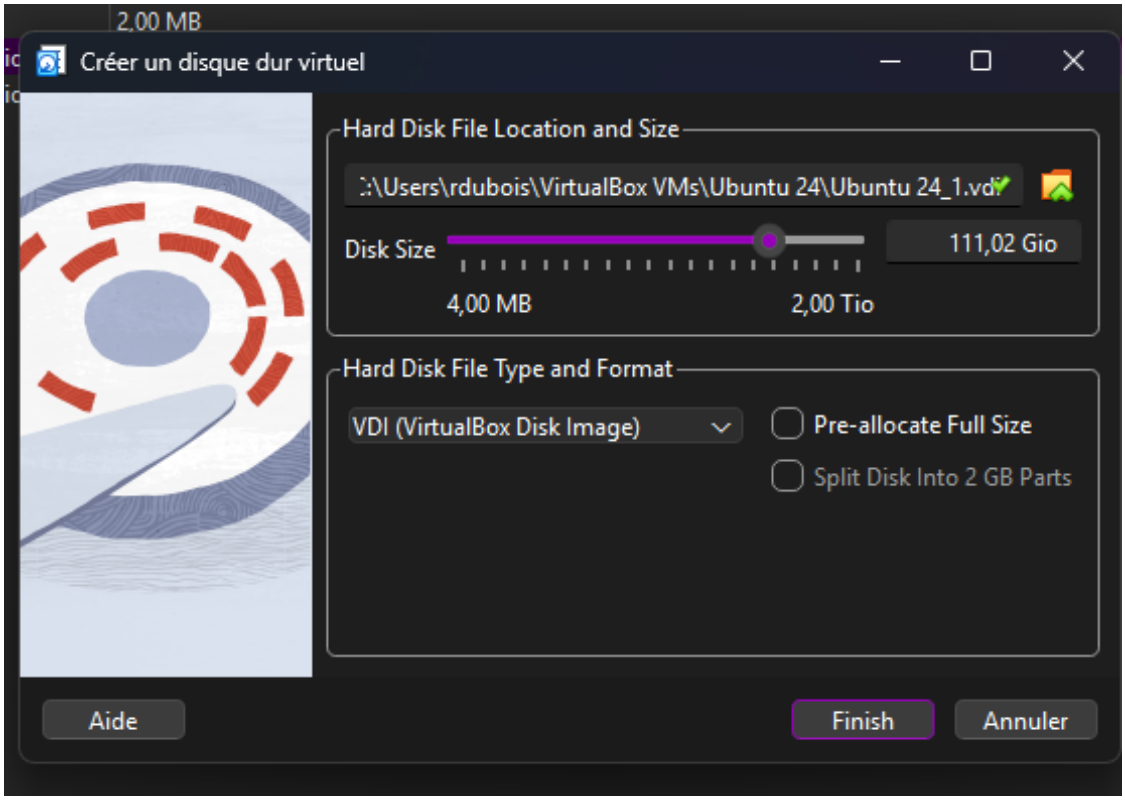
4) Tester la distribution RedoBackup

Avant de démarrer sur RedoBackup, nous allons créer un nouveau disque sur notre VM pour sauvegarder les données de cette dernière sur un nouveau disque, pour cela il faut se rendre dans les paramètres de cette dernière, puis dans Stockage :

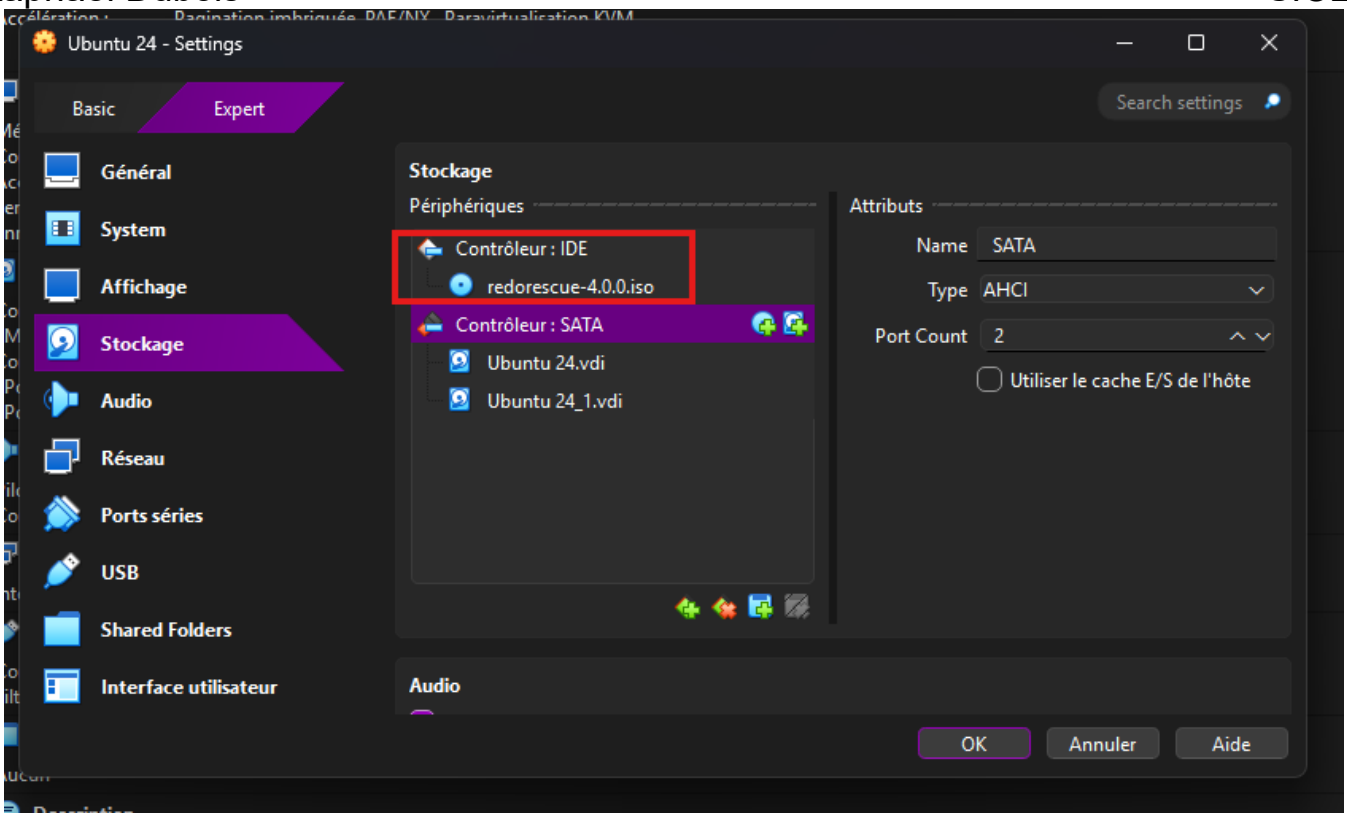


Nous allons donc créer un nouveau disque ayant la même taille que notre lecteur actuel :

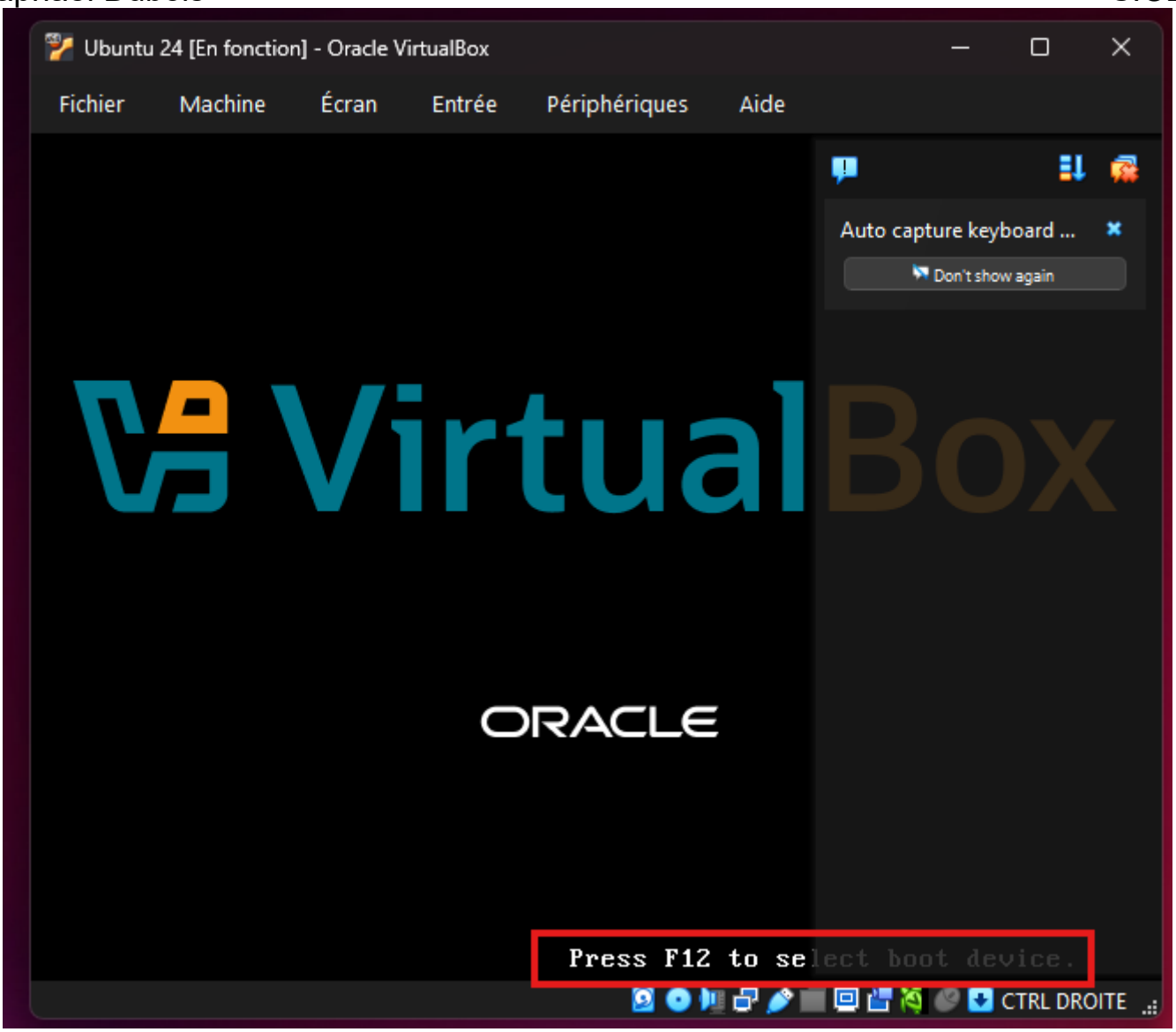




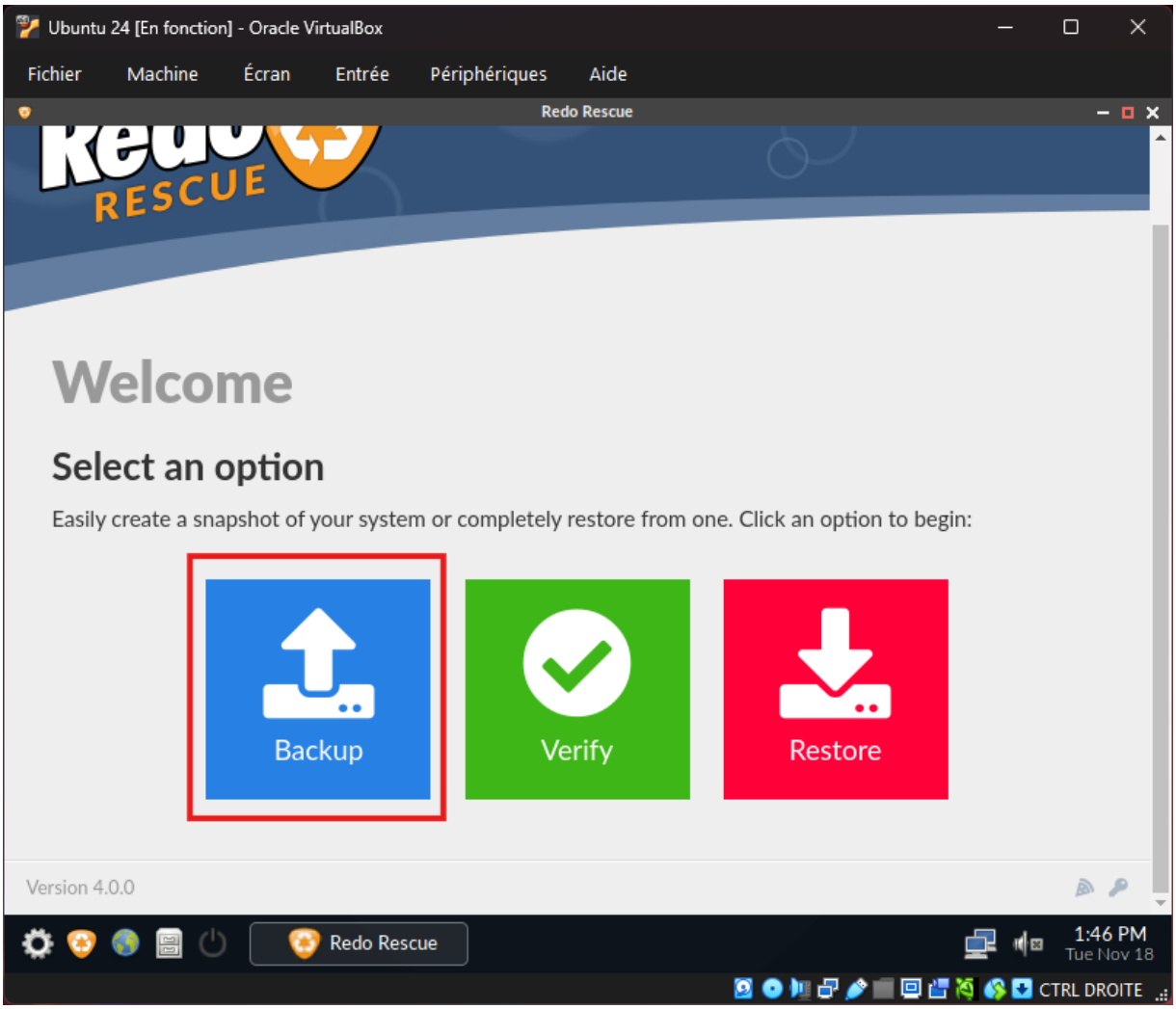
Enfin, une fois que le disque a été créer, nous allons ajouter l'image ISO de RedoBackup :



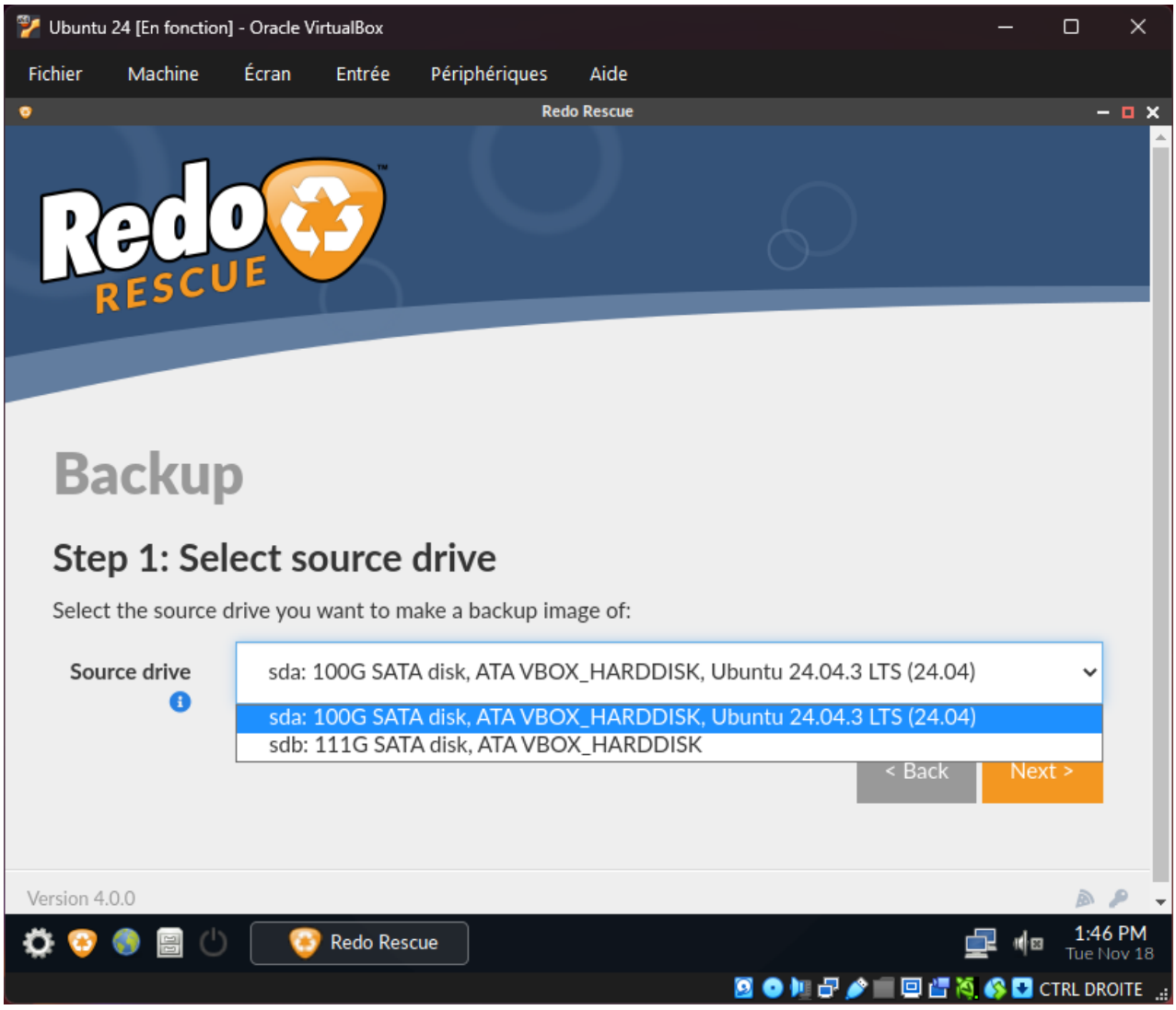
Une fois ceci fait, nous pouvons ajouter l'iso de RedoBackup sur notre VM, puis appuyer sur F12 au démarrage de la VM pour booter sur le disque RedoBackup :



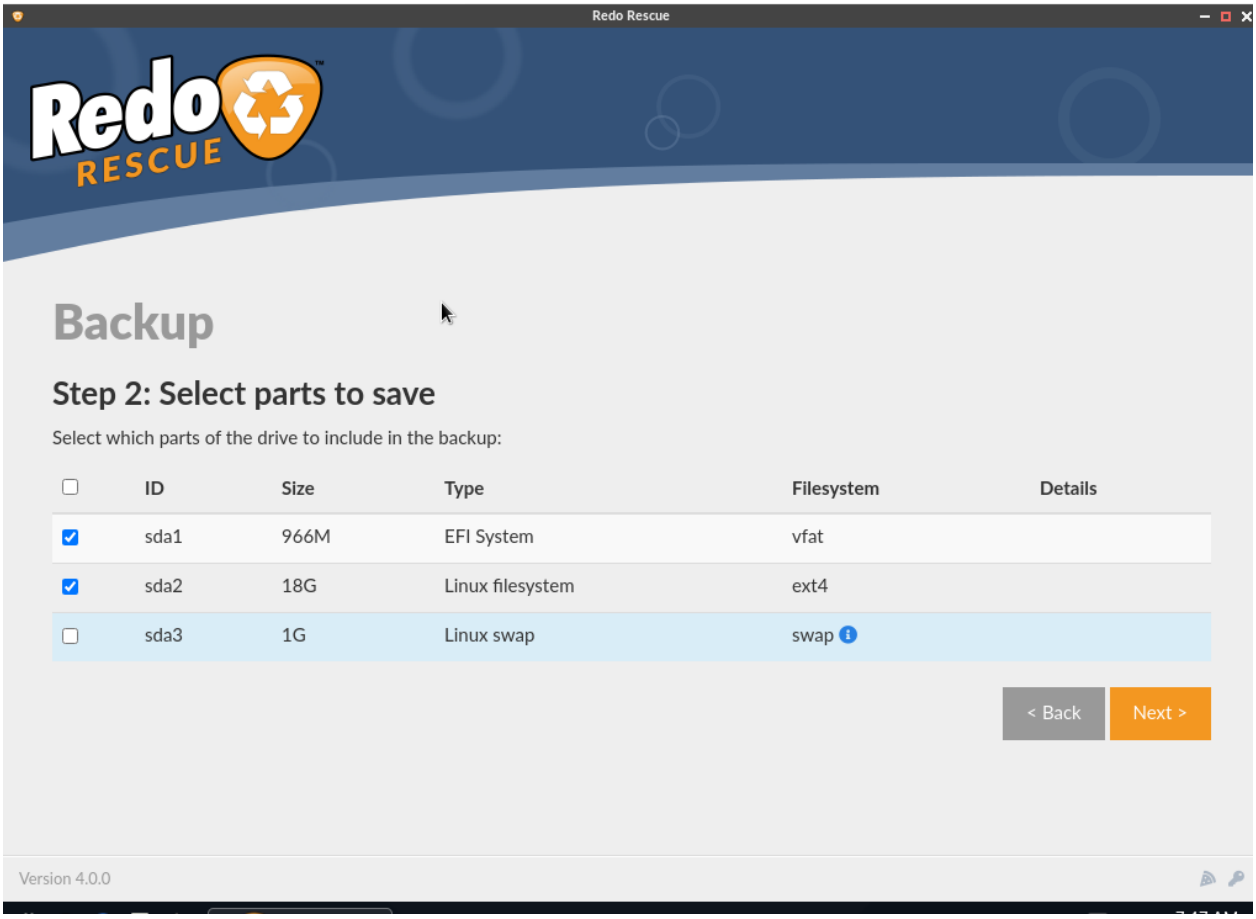
Une fois dans RedoBackup, nous allons cliquer sur « Backup » pour faire une sauvegarde de notre VM :



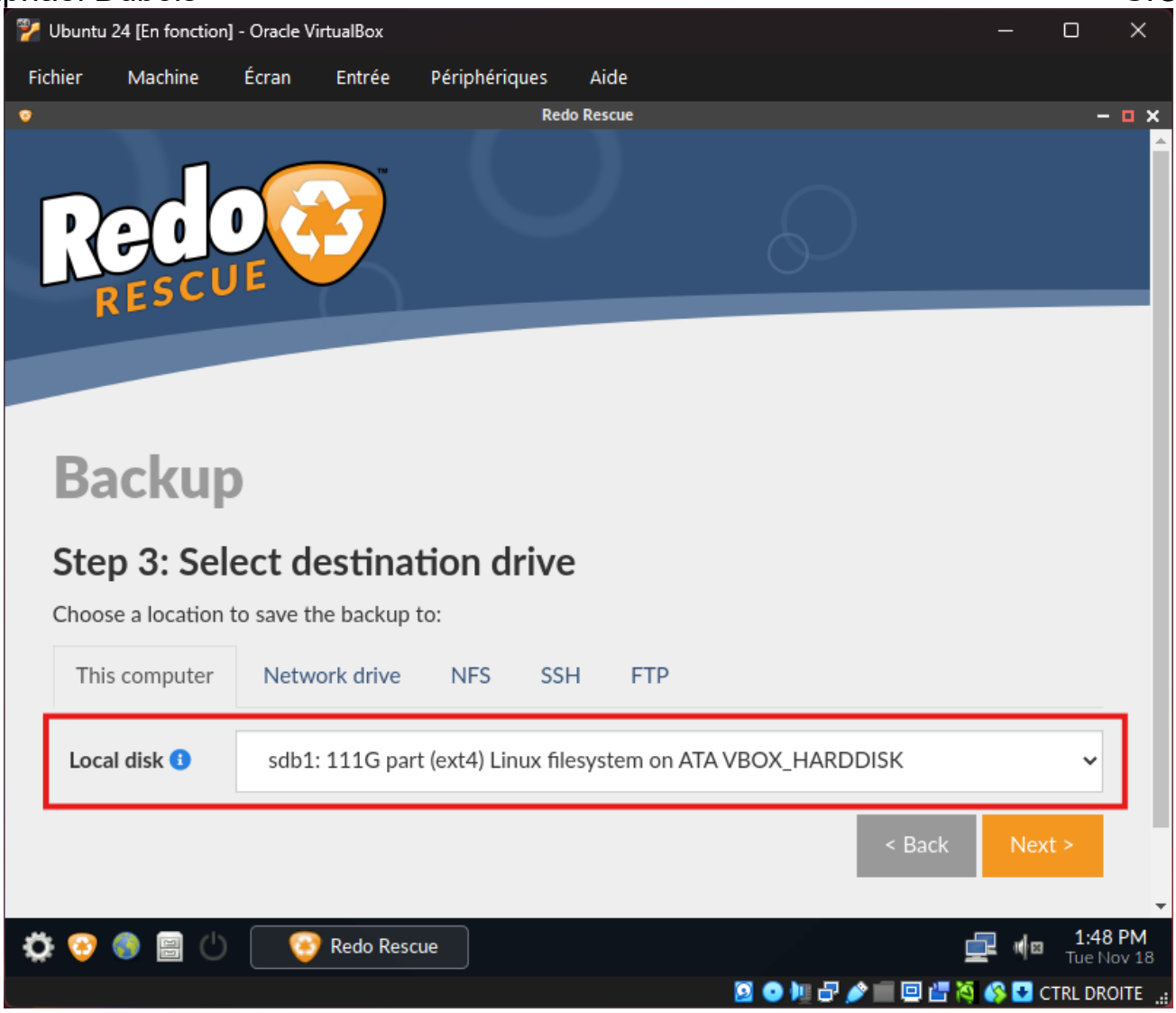
Nous allons ensuite sélectionner le disque source (le disque à sauvegarder) :



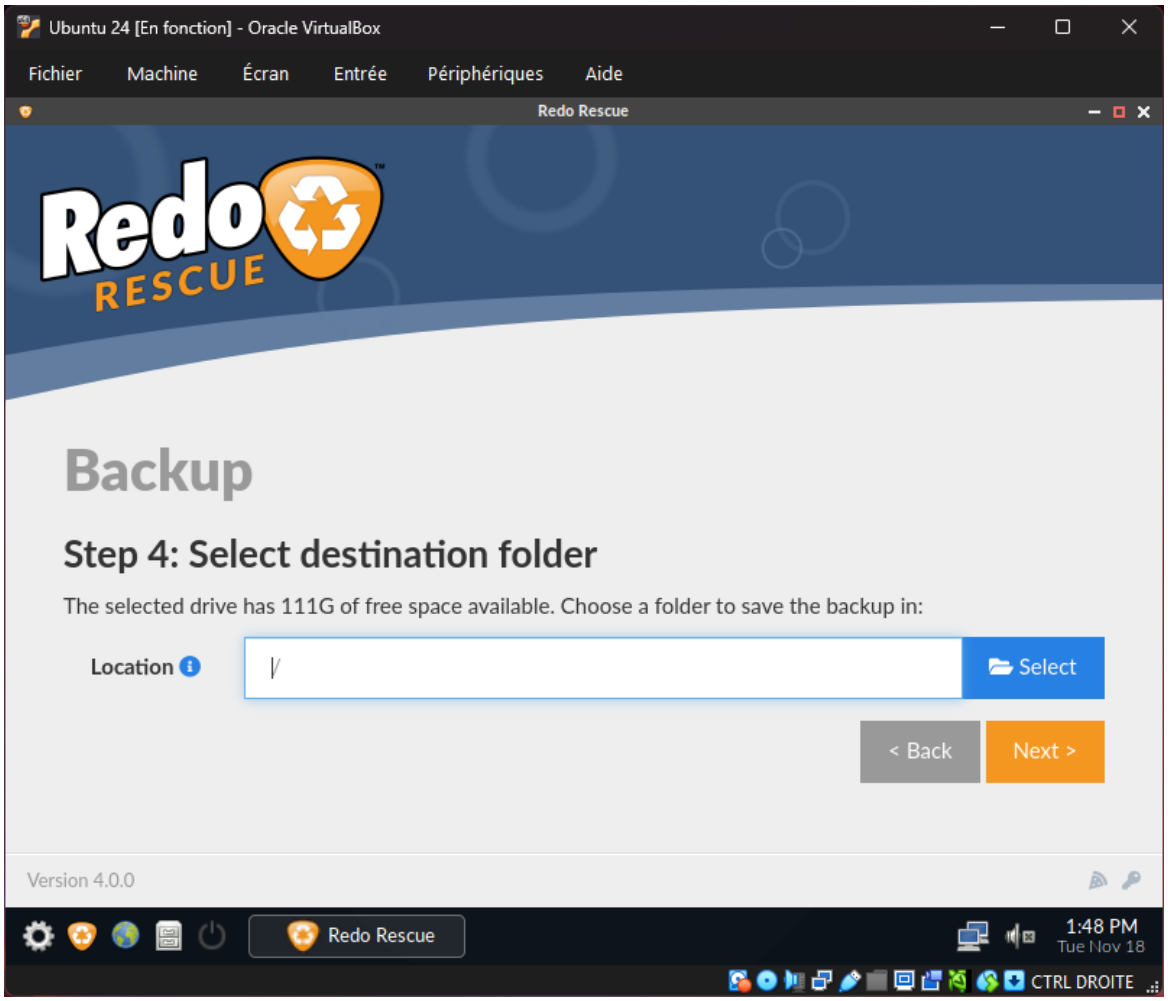
Ensuite, nous allons sélectionner les partitions à sauvegarder (en l'occurrence, toutes sauf le Swap) :



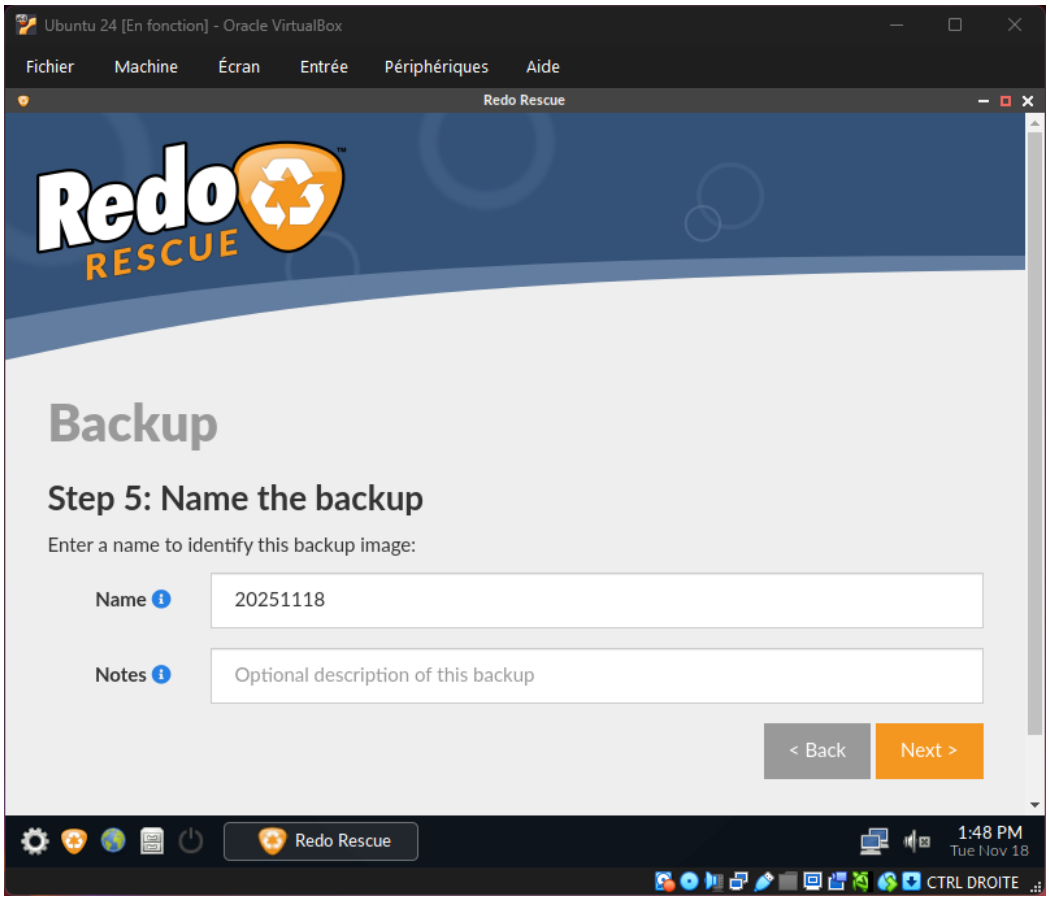
Ensuite, nous allons sélectionner le disque cible (le disque qui va recevoir et stocker la sauvegarde) :

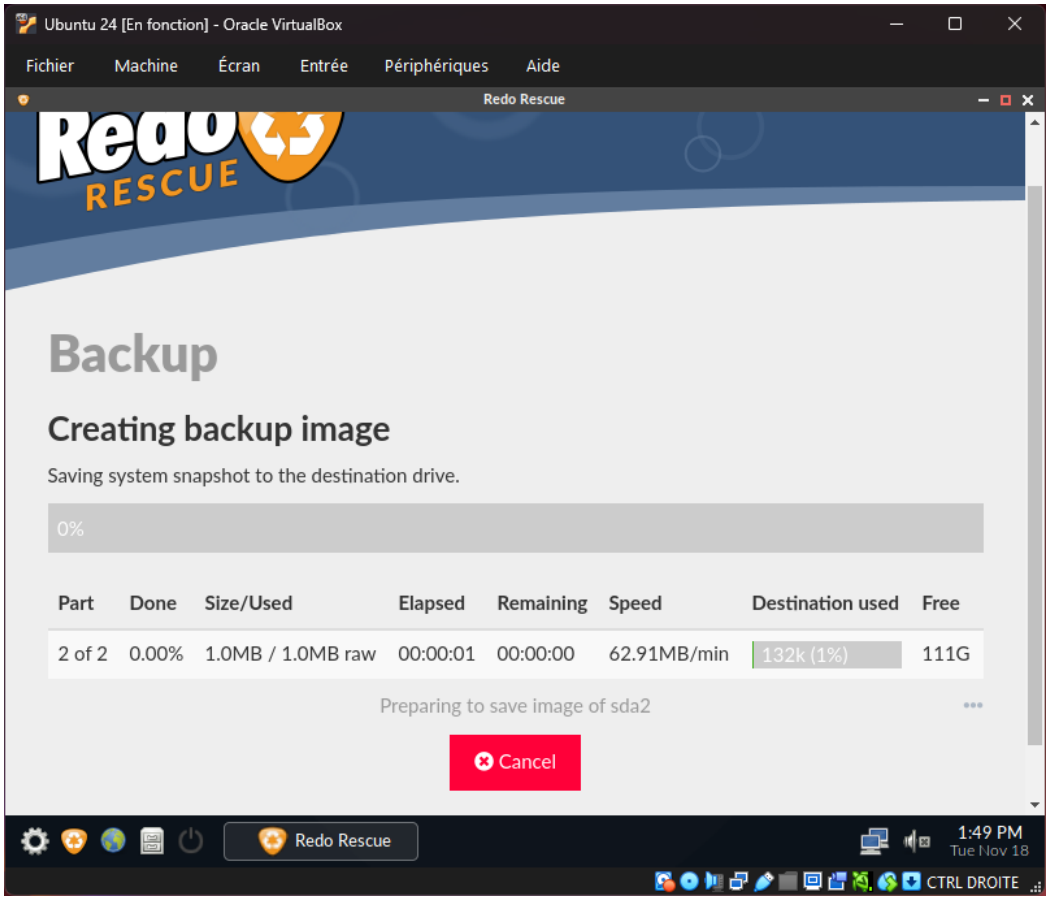


Puis, nous allons sélectionner le répertoire ou sauvegarder les données (en l'occurrence, nous allons garder celui par défaut étant la racine du disque) :

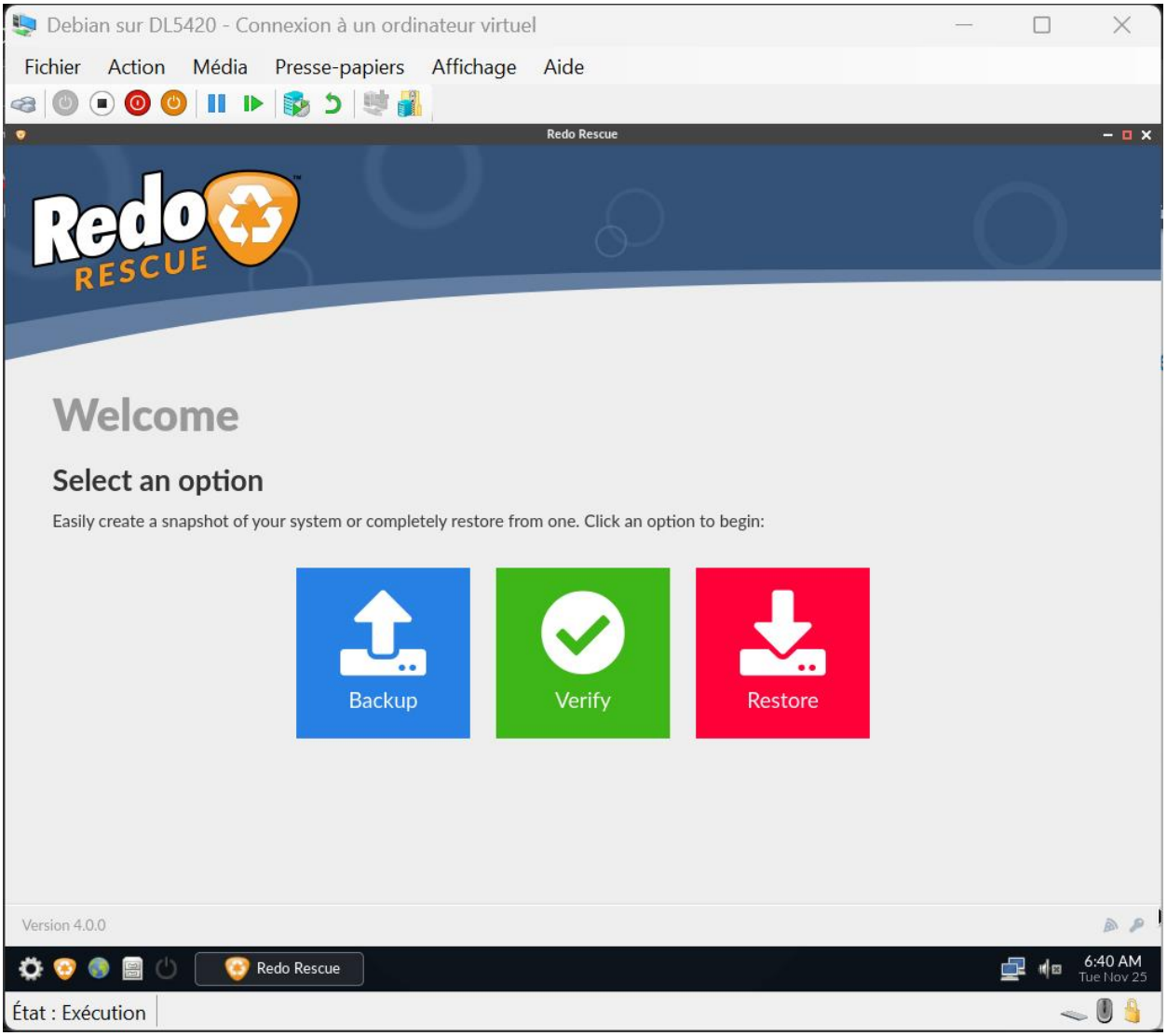


Et puis enfin, dernière étape avant de lancer la sauvegarde, nous allons définir un nom à cette dernière (par défaut : Année/Mois/Jour) :

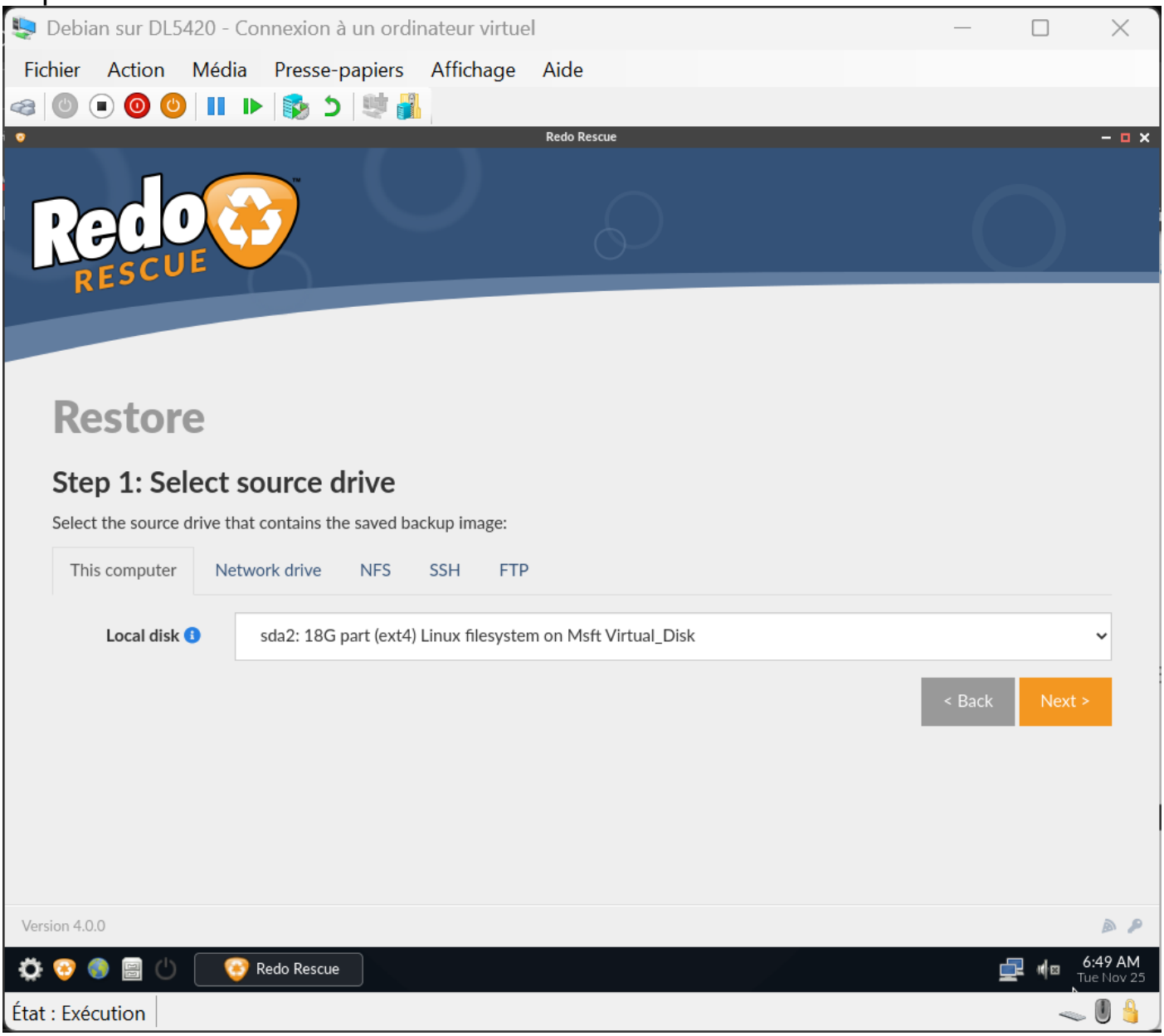

Ubuntu 24 [En fonction] - Oracle VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Redo Rescue
Redo RESCUE
Backup
Step 5: Name the backup
Enter a name to identify this backup image:
Name
20251118
Notes
Optional description of this backup
< Back
Next >
Redo Rescue
1:48 PM
Tue Nov 18
CTRL DROITE


Ubuntu 24 [En fonction] - Oracle VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Redo Rescue
Redo RESCUE
Backup
Creating backup image
Saving system snapshot to the destination drive.
0%
Part Done Size/Used Elapsed Remaining Speed Destination used Free
2 of 2 0.00% 1.0MB / 1.0MB raw 00:00:01 00:00:00 62.91MB/min 132k (1%) 111G
Preparing to save image of sda2
Cancel
Redo Rescue
1:49 PM
Tue Nov 18
CTRL DROITE

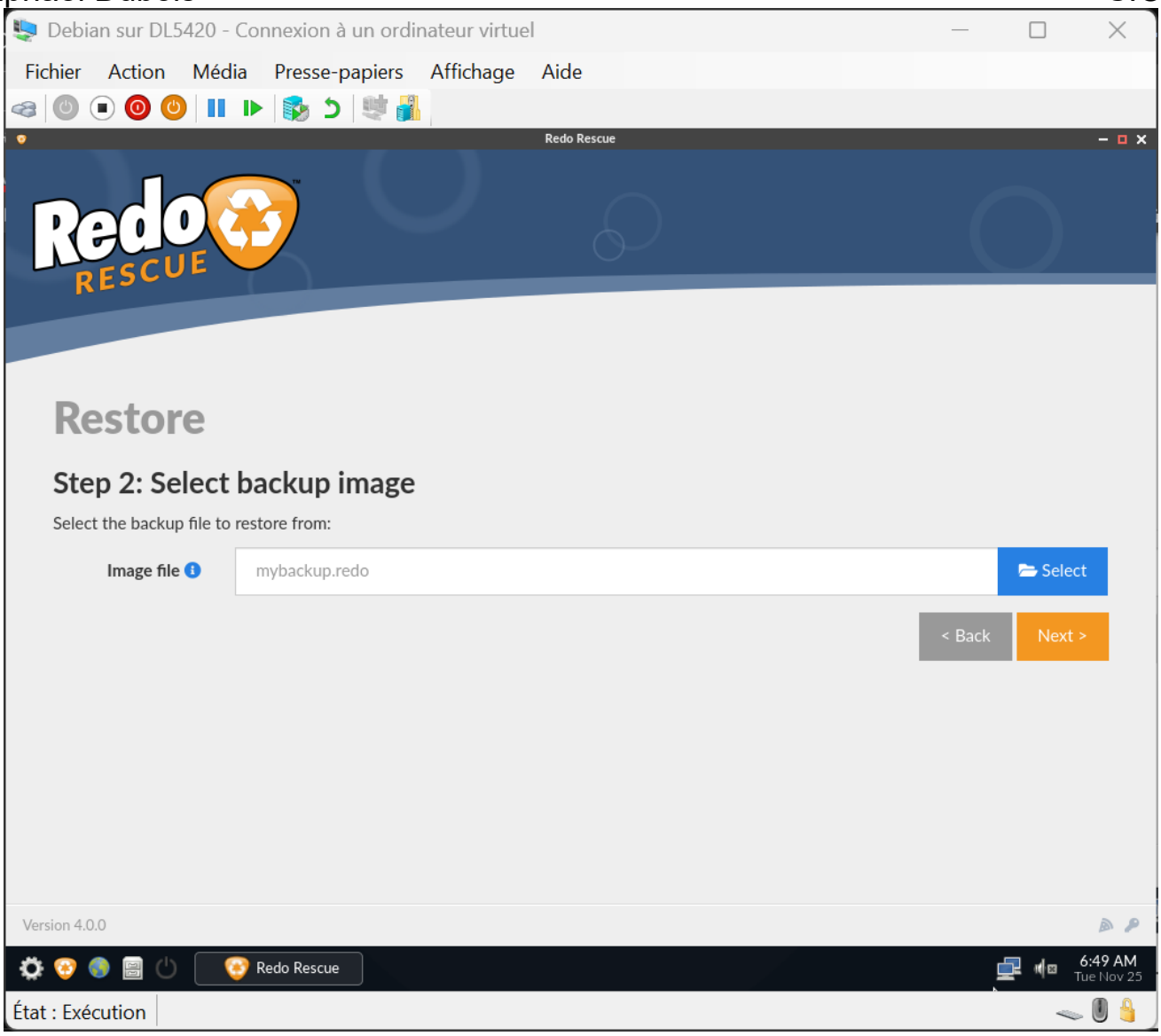
Une fois la sauvegarde terminée, nous allons tenter de restaurer cette dernière sur notre disque principal, pour cela, nous allons lancer RedoBackup en mode « Restore » :



Nous allons ensuite sélectionner le disque contenant notre sauvegarde précédemment créer :



Une fois ceci fait, il suffit de sélectionner le fichier, de confirmer et de patienter quelques minutes le temps que les données soient restaurées :



Une fois que la restauration est terminée, on peut redémarrer la VM sur le disque pour contrôler que tout a bien fonctionné et que notre VM puisse toujours démarrer.

5) Synchronisation avec Unison

1) Linux

PARTIE 1 : Utilisation de Unison GTK

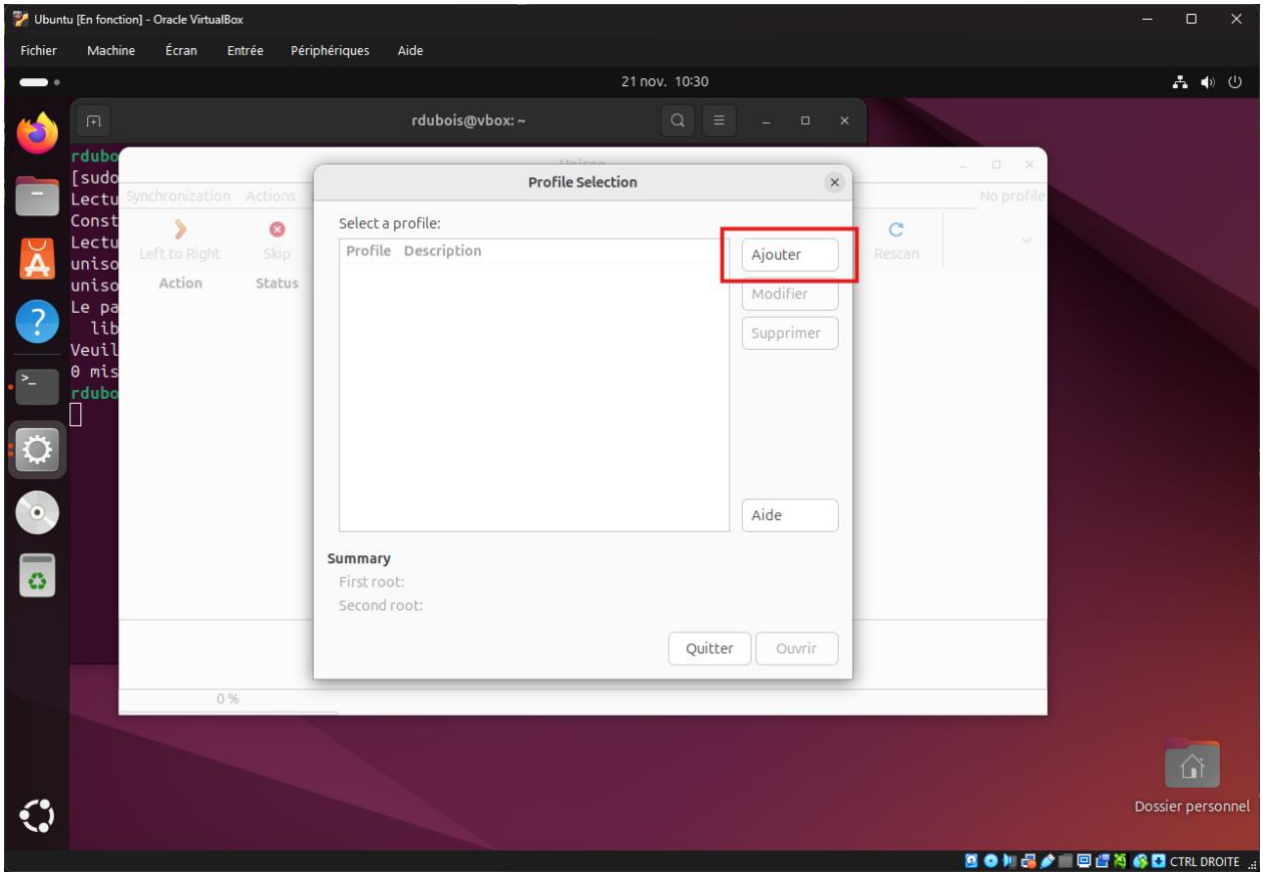
Premièrement, pour installer Unison sur Linux, nous devons ouvrir un terminal puis exécuter les commandes suivantes :

```
sudo apt update
```

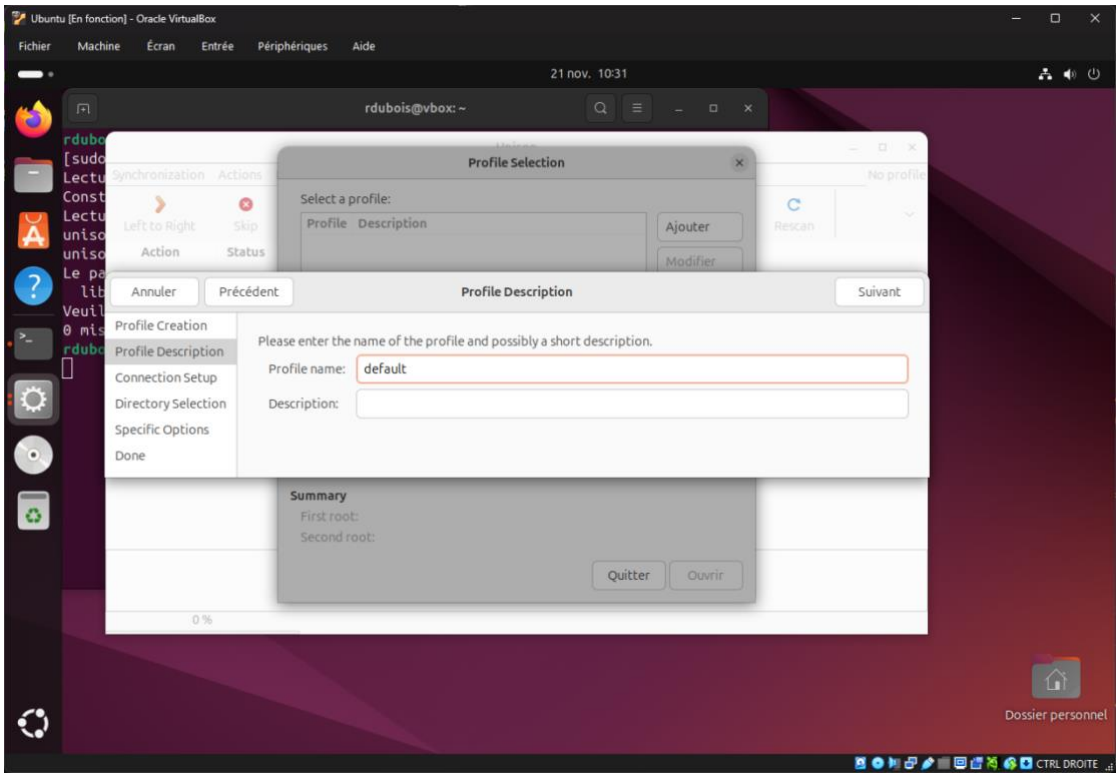
```
sudo apt install unison unison-gtk
```

Le paquet unison étant la version en ligne de commande, et unison-gtk la version graphique.

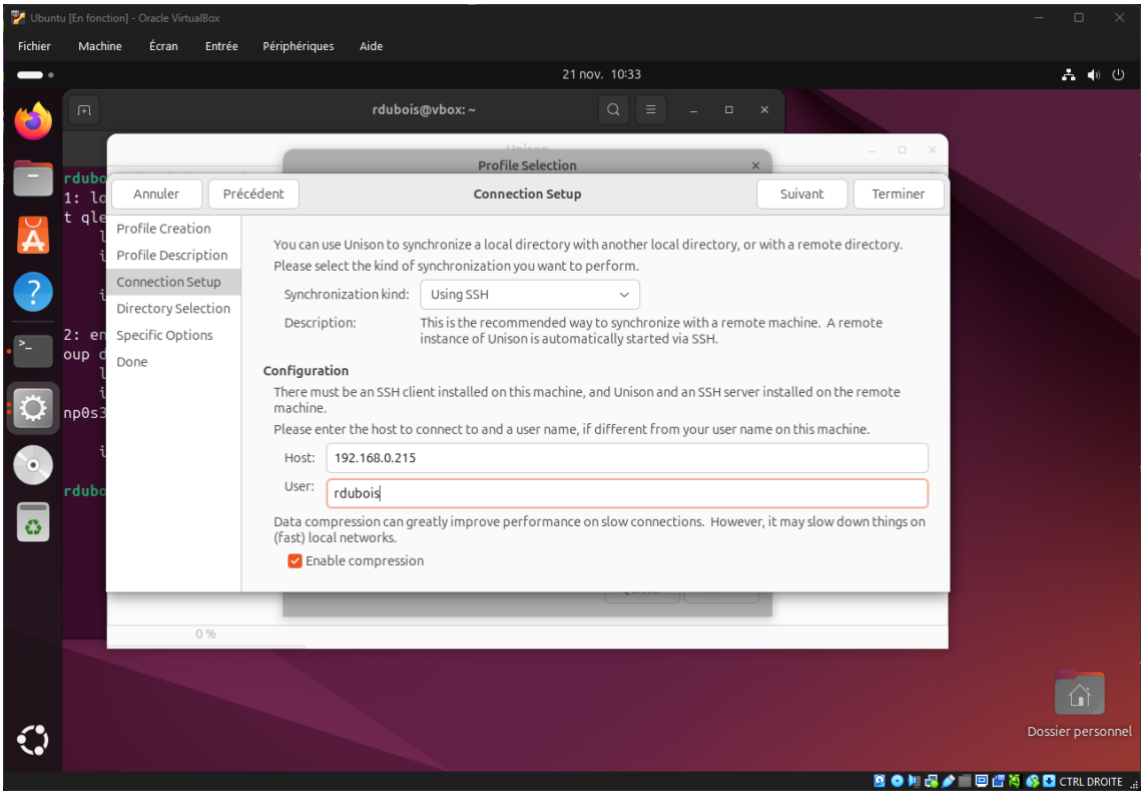
Une fois ceci fait, nous pouvons lancer dans notre terminal, la commande « unison-gtk » qui va exécuter la version graphique de Unison :



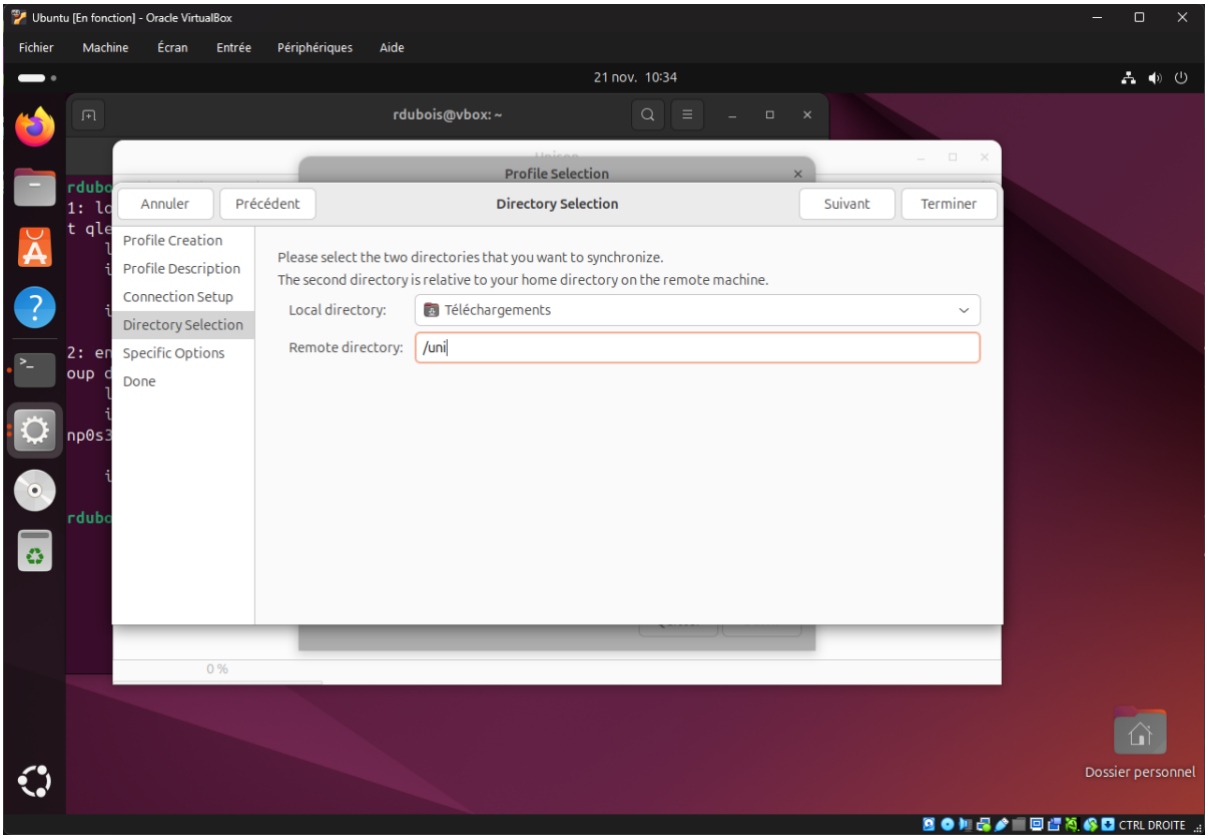
A partir de là, nous allons créer un nouveau profil appelé « default » :



Puis dans la suite du processus, nous allons faire une synchronisation en SSH avec notre autre VM Linux :

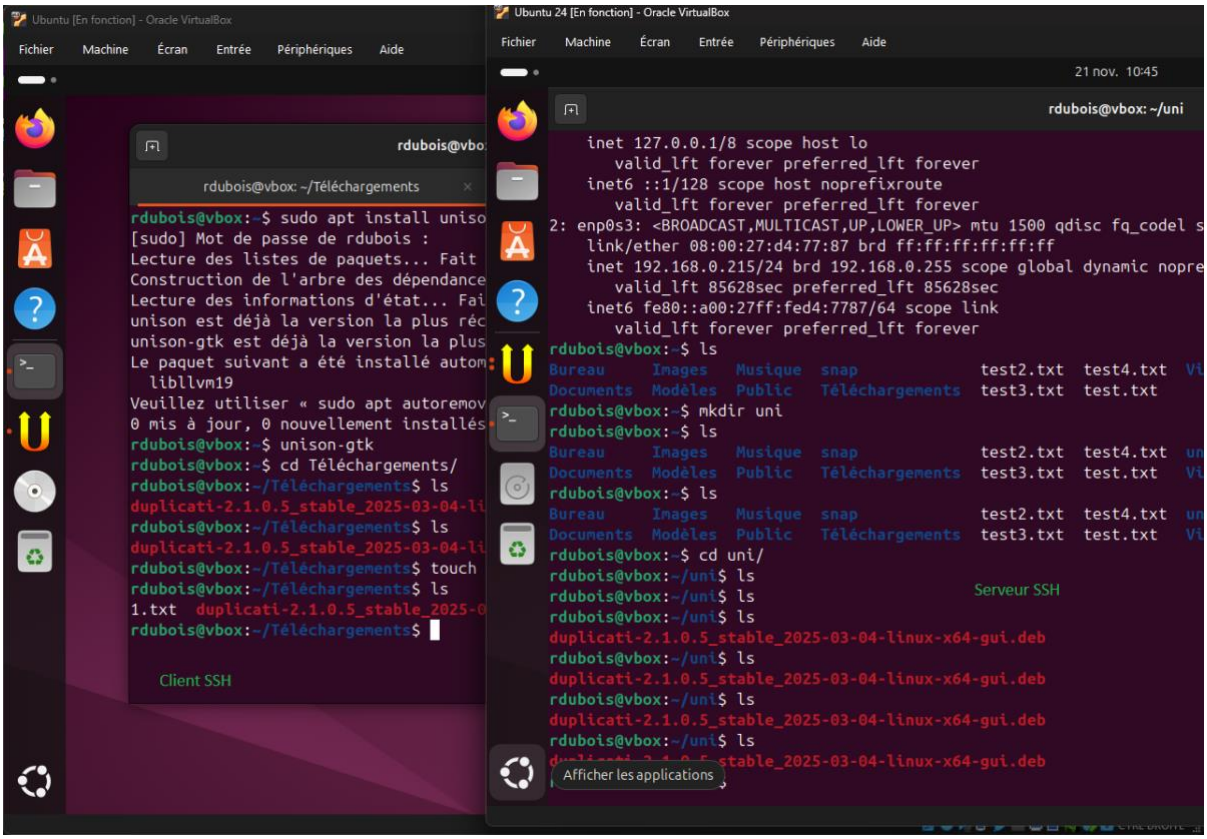


Le dossier synchronisé sera /home/rdubois/uni, il sera créé en amont sur la VM avec le serveur SSH (le client synchronisera son dossier /home/rdubois/Téléchargements avec le dossier uni du serveur) :

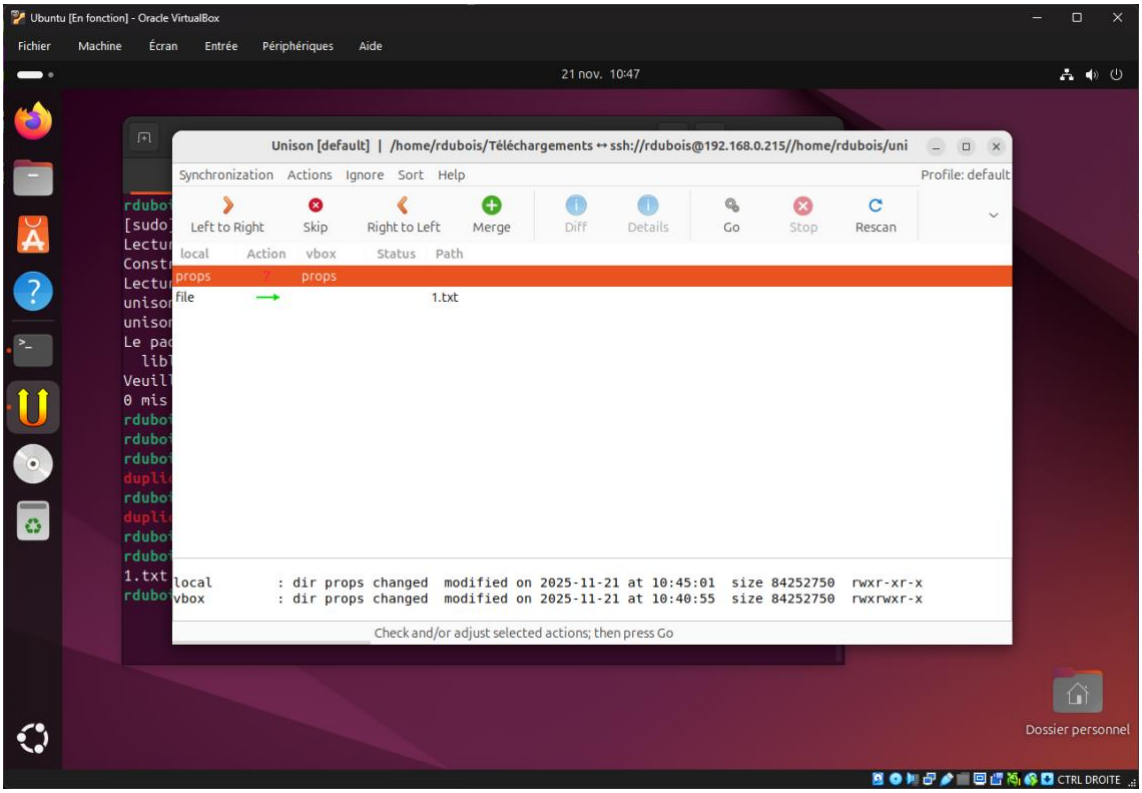


Une fois que la configuration est terminée, Unison demandera le mot de passe du compte rdubois pour se connecter au serveur, il est enregistré tant que Unison est ouvert et devra être ré indiqué en cas de redémarrage de Unison (**IMPORTANT**, il faut que unison soit aussi installé sur le serveur SSH, même si on ne va pas l'utiliser sur ce dernier).

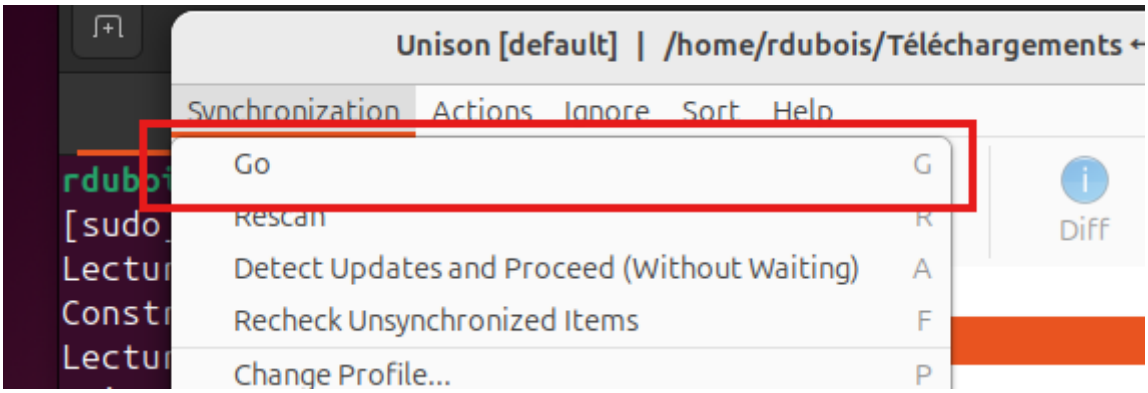
Voici notre configuration actuelle, d'un côté nous avons le client SSH, et de l'autre le serveur, sur le client, nous avons créé un fichier « 1.txt », qui pour le moment n'est pas encore synchronisé avec le serveur :



Nous allons donc démarrer la synchronisation, pour cela dans Unison, nous allons premièrement cliquer sur « Rescan » :



Nous avons notre fichier 1.txt qui apparait, pour le synchroniser, il suffit d'aller dans « Syncronization » et de cliquer sur « Go » :



Nous pouvons maintenant vérifier le serveur et remarquer que le fichier 1.txt a bien été synchronisé entre le client et le serveur !

PARTIE 2 : Automatisation avec un service systemd :

Avant de créer un service, nous allons créer une clé SSH puis l'importer sur le serveur afin de pouvoir automatiser la connexion sans avoir à y insérer de mot de passe :

```
user@debian:~$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/user/.ssh/id_ed25519):
Enter passphrase for "/home/user/.ssh/id_ed25519" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_ed25519
Your public key has been saved in /home/user/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:O0aYFcFWm7xMhlrVh+d8bAg1mB7A9TnNThdj78VI9R4 user@debian
The key's randomart image is:
+--[ED25519 256]--+
|        .o+++.=+=.|
|         o=.oBo+B=|
|        .+ *. B=EX|
|        * + .. +*B|
|       + S o    o+|
|        . .       |
|         +        |
|        . .       |
|                  |
+----[SHA256]-----+
user@debian:~$
```

Nous allons maintenant copier la clé publique sur le serveur avec la commande suivante :

```
ssh-copy-id -i ~/.ssh/id_ed25519.pub user@ip_server
```

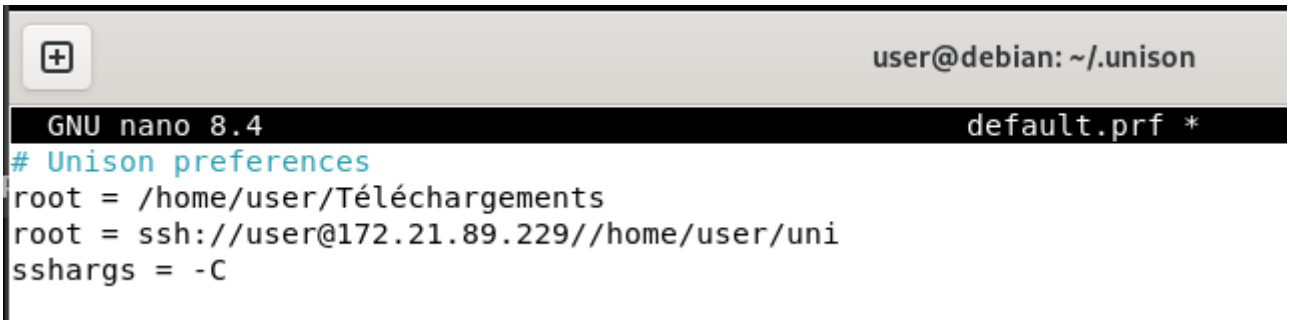
```
user@debian:~$ ssh-copy-id -i ~/.ssh/id_ed25519.pub user@172.21.89.229
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/user/.ssh/i
d_ed25519.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
user@172.21.89.229's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh -i /home/user/.ssh/id_ed25519 'user
@172.21.89.229'"
and check to make sure that only the key(s) you wanted were added.

user@debian:~$
```

Maintenant, si l'on essaie de lancer une synchronisation avec la commande « unison », aucun mot de passe ne sera demandé :

```
user@debian:~$ unison
Unison 2.53.7 (ocaml 5.2.0): Contacting server...
Connected [//debian//home/user/Téléchargements -> //debian2//home/user/uni]

Looking for changes
  Waiting for changes from server
Reconciling changes

local          debian2
props    <-?-> props        [] -
new file ---->            2.txt  [f]

1 items will be synced, 0 skipped
0 B to be synced from local to debian2
0 B to be synced from debian2 to local

Proceed with propagating updates? []
No default command [type '?' or F1 for help]
Proceed with propagating updates? [] y
Propagating updates


Unison 2.53.7 (ocaml 5.2.0) started propagating changes at 09:09:05.21 on 25 Nov 2025
[BGN] Copying 2.txt from /home/user/Téléchargements to //debian2//home/user/uni
Shortcut: copied /home/user/uni/2.txt from local file /home/user/uni/1.txt
[END] Copying 2.txt
Unison 2.53.7 (ocaml 5.2.0) finished propagating changes at 09:09:05.22 on 25 Nov 2025, 0.008 s


Saving synchronizer state
Synchronization complete at 09:09:05  (1 item transferred, 0 skipped, 0 failed)
user@debian:~$
```

Maintenant, nous allons faire en sorte que Unison ne nous demande pas de confirmer dans le terminal les fichiers que l'on souhaite synchroniser, pour cela, nous allons ouvrir le fichier suivant avec nano : ~/.unison/default.prf

Ce fichier contient les paramètres de notre profil default :



Pour automatiser la commande, nous allons ajouter ces deux lignes dans le fichier :

```
auto = true
```

```
batch = true
```

Une fois les changements effectués, on peut sauvegarder le fichier, et tenter d'exécuter la commande Unison, cette fois ci, Unison ne demande pas de confirmation pour la synchronisation :

```
user@debian:~/.unison$ unison
Unison 2.53.7 (ocaml 5.2.0): Contacting server...
Connected [//debian//home/user/Téléchargements -> //debian2//home/user/uni]

Looking for changes
  Waiting for changes from server
Reconciling changes
props    <-?-> props
local        : dir props changed  modified on 2025-11-25 at  9:08:43  size 0         rwxr-xr-x
debian2      : dir props changed  modified on 2025-11-25 at  9:09:05  size 0         rwxrwxr-x

0 items will be synced, 1 skipped
0 B to be synced from local to debian2
0 B to be synced from debian2 to local
No updates to propagate
Synchronization complete at 09:13:11  (0 items transferred, 1 skipped, 0 failed)
  skipped:  (properties changed on both sides)
user@debian:~/.unison$
```

Maintenant, nous allons créer un service systemd qui va s'exécuter toute les 5 minutes pour lancer la commande unison (et donc synchroniser les fichiers).

Pour cela, nous allons nous rendre dans le dossier ~/.config/systemd/user et nous allons créer deux fichiers : **unison-default.service** ; **unison-default.timer**

Le fichier service va contenir la commande à exécuter, tandis que le fichier timer va plutôt contenir la fréquence d'exécution :

user@debian: ~/.config/systemd/user

```
 GNU nano 8.4                          unison-default.service
[Unit]
# Description courte du service
Description=Syncro Unison automatique

[Service]
# S'arrête après l'execution du script
Type=oneshot
# Commande a executer
ExecStart=/usr/bin/unison default -batch
```

user@debian: ~/.config/systemd/user

```
 GNU nano 8.4                          unison-default.timer
[Unit]
# Description du timer
Description=Démarrer Unison toute les 5 minutes

[Timer]
# Lance le service après 1 minute au démarrage
OnBootSec=1m
# Après le premier démarrage, le service se lance toutes les 5 minutes
OnUnitActiveSec=5m
# Lancer le service unison-default.service (précèdement créer)
Unit=unison-default.service

[Install]
# Nécessaire au fonctionnement du timer
WantedBy=timers.target
```

Une fois nos fichiers créer, on peut utiliser les commandes suivantes pour activer le timer :

```
systemctl --user enable unison-default.timer

systemctl --user start unison-default.timer
```

```
user@debian:~/.config/systemd/user$ systemctl --user enable unison-default.timer
Created symlink '/home/user/.config/systemd/user/timers.target.wants/unison-default.timer' → '/home/user/.config
/systemd/user/unison-default.timer'.
```

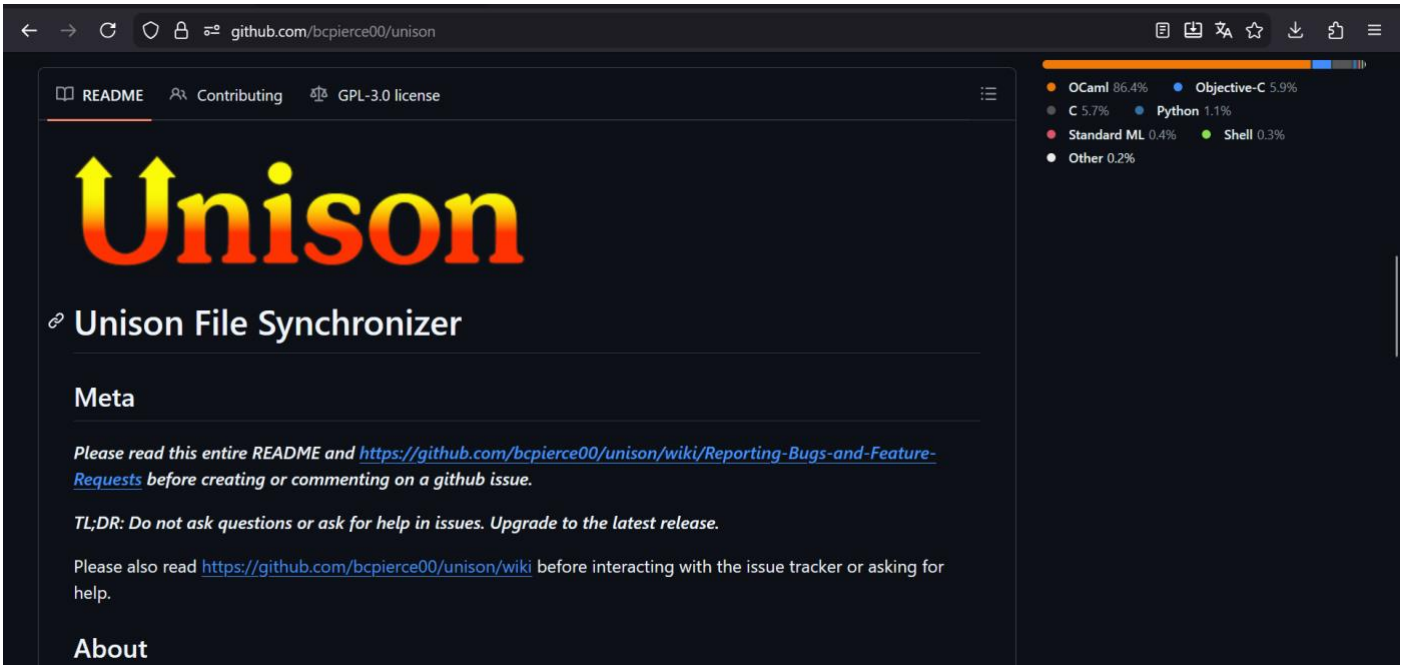
```
user@debian:~/.config/systemd/user$ systemctl --user start unison-default.timer
user@debian:~/.config/systemd/user$ systemctl --user status unison-default.timer
● unison-default.timer - Démarrer Unison toute les 5 minutes
     Loaded: loaded (/home/user/.config/systemd/user/unison-default.timer; enabled; preset: enabled)
     Active: active (waiting) since Tue 2025-11-25 09:33:06 CET; 1s ago
 Invocation: 475c72599bcb49d39a2f8610c7a7705b
    Trigger: Tue 2025-11-25 09:38:06 CET; 4min 58s left
   Triggers: ● unison-default.service

nov. 25 09:33:06 debian systemd[1462]: Started unison-default.timer - Démarrer Unison toute les 5 minutes.
```

Avec la commande status, on peut aussi vérifier le temps restant avant la prochaine exécution du service, pour vérifier si cela fonctionne, nous allons créer un fichier 3.txt et vérifier s'il se synchronise bien automatiquement :

Vidéo de démonstration : https://drive.proton.me/urls/07422G7Y30#9iBSO0jiRAvS

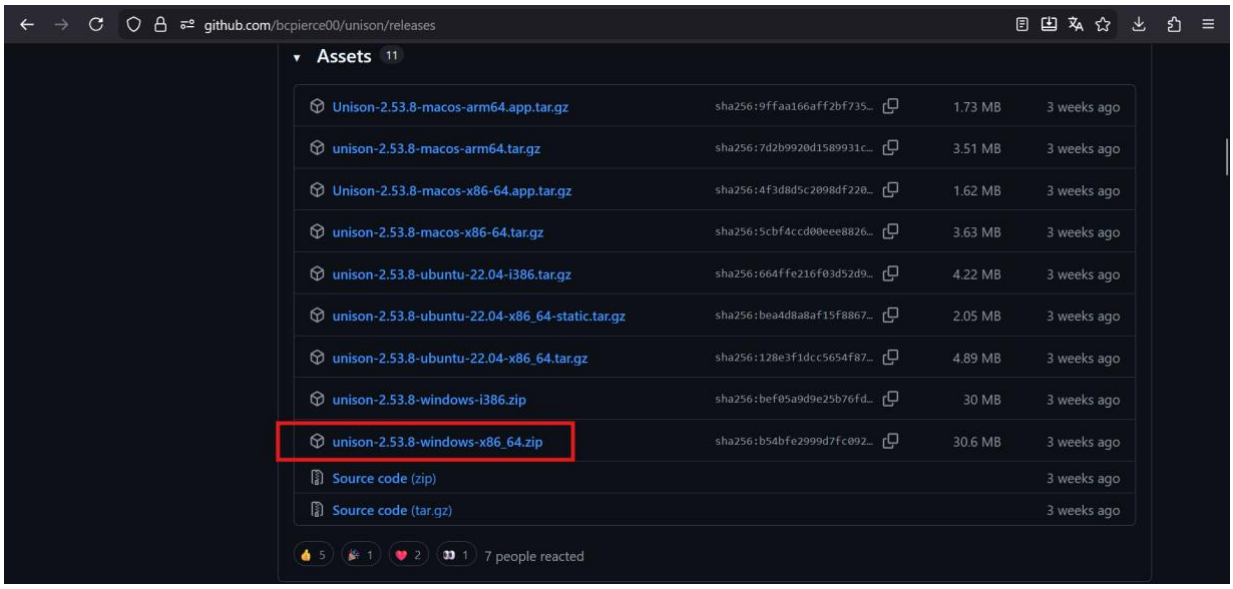
2) Windows

Pour télécharger la version Windows, nous allons nous rendre sur la page officiel Github de Unison :

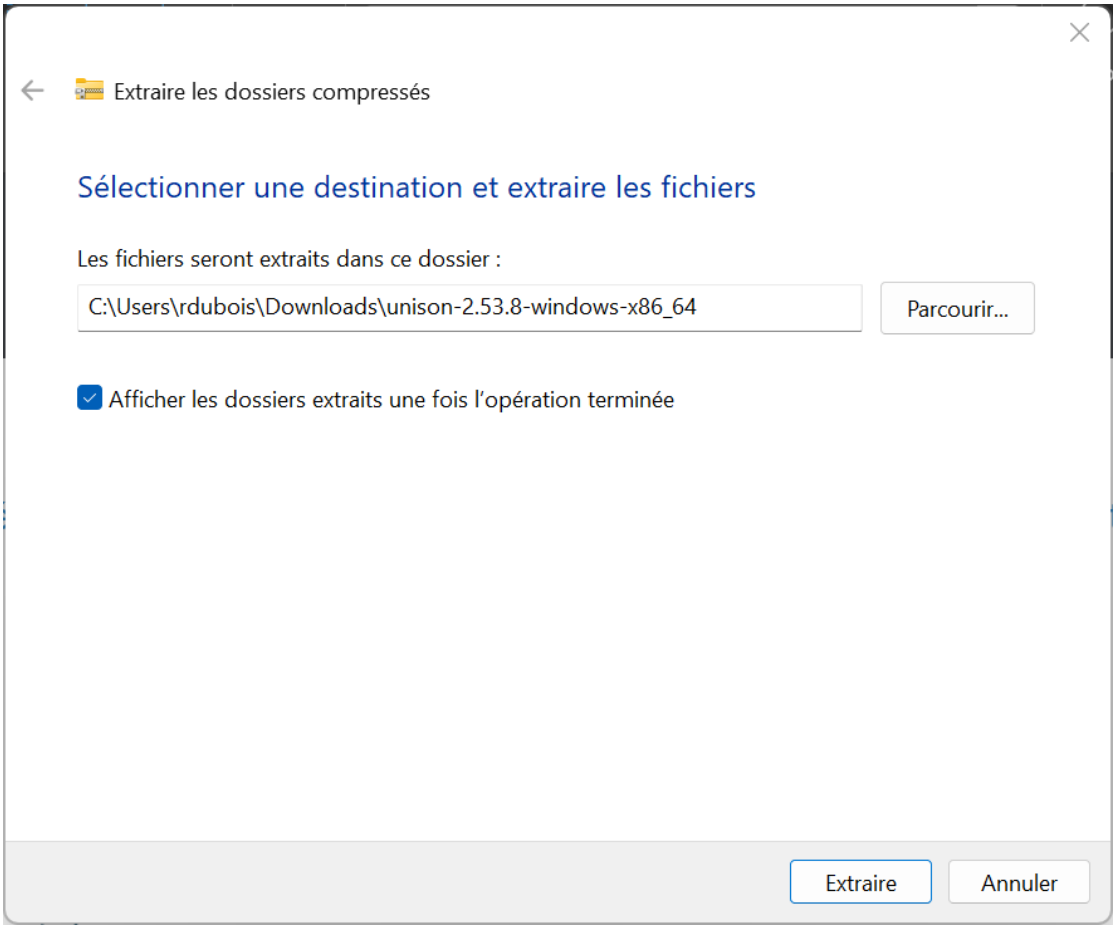


Une fois sur cette page, nous allons aller dans les « Releases » pour télécharger la dernière version d'Unison pour Windows :

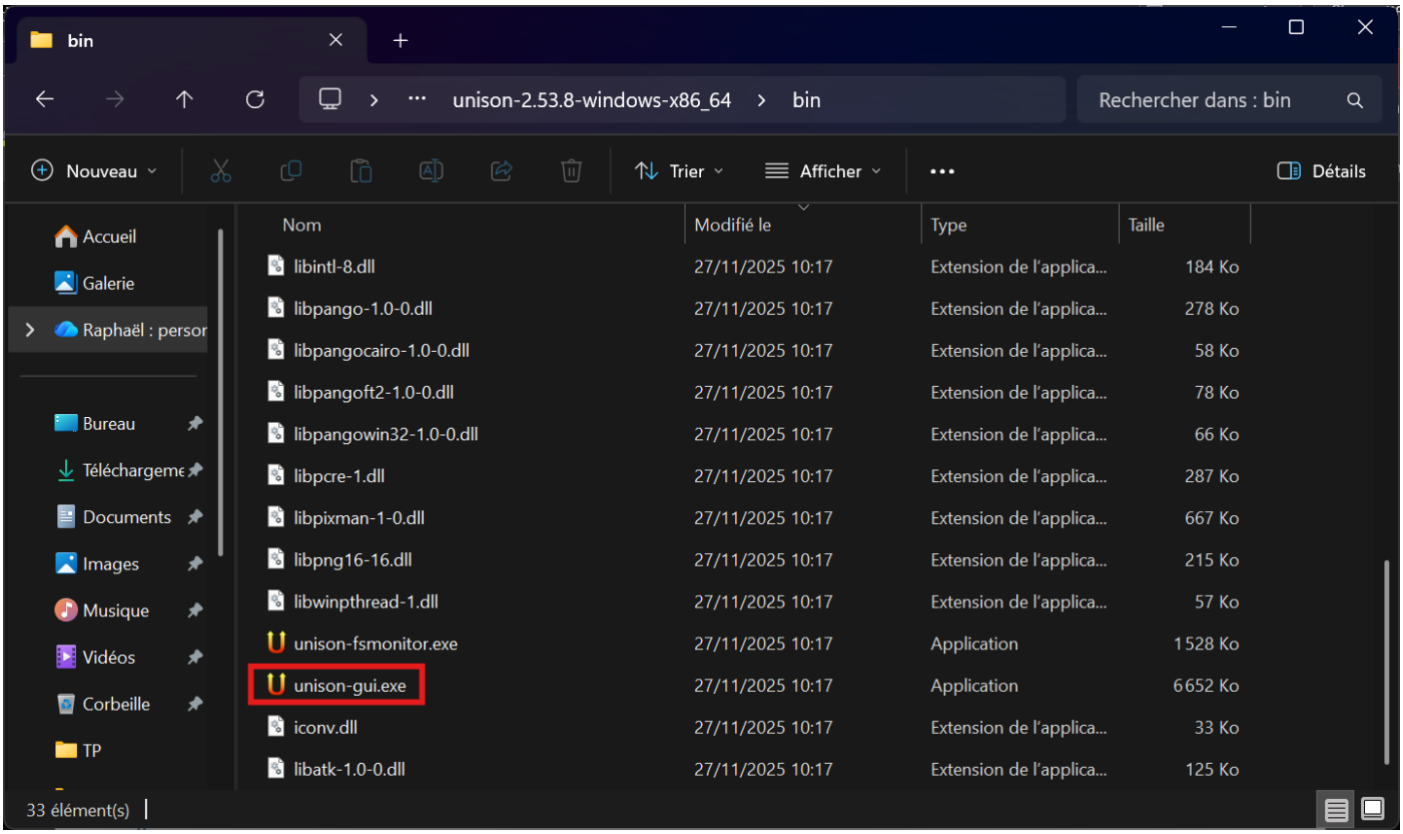




Une fois téléchargé, nous pouvons extraire l'archive en faisant clic droit et extraire tout :



Une fois extrait, il suffit de lancer Unison, pour cela, il suffit d'aller dans le dossier Unison, bin, puis unison-gui.exe :

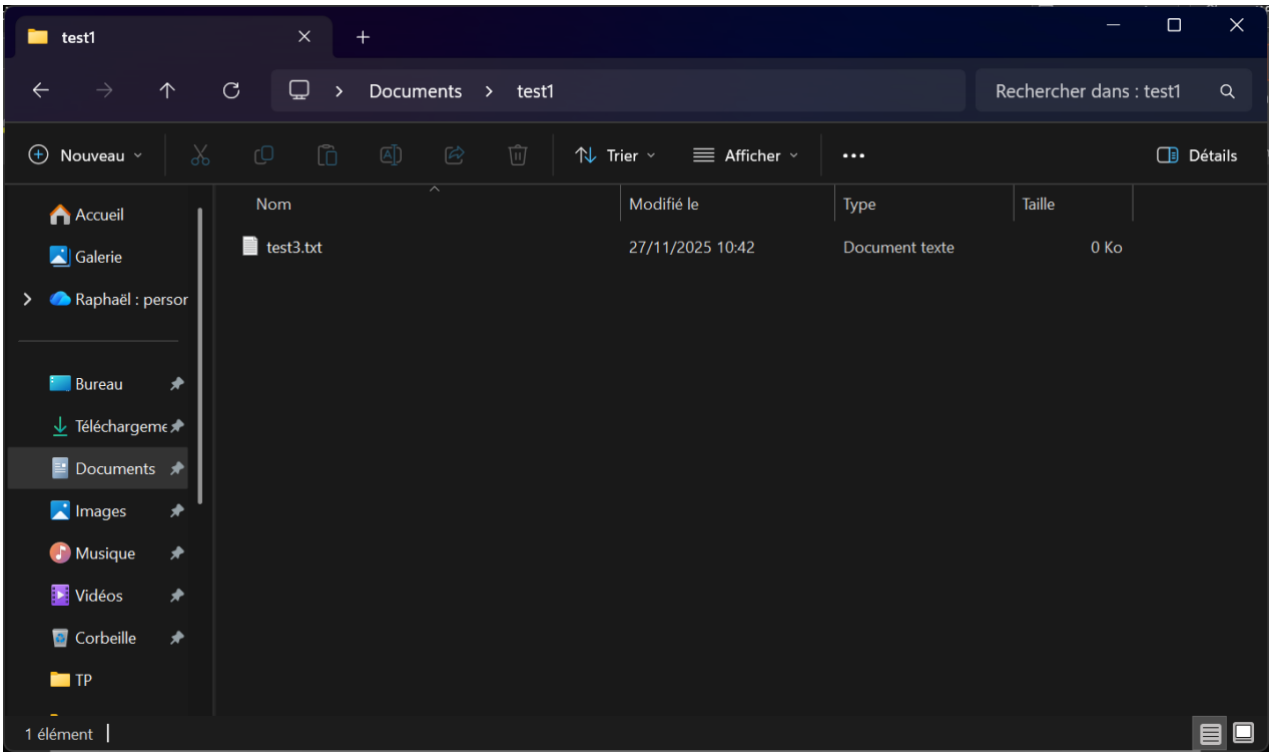


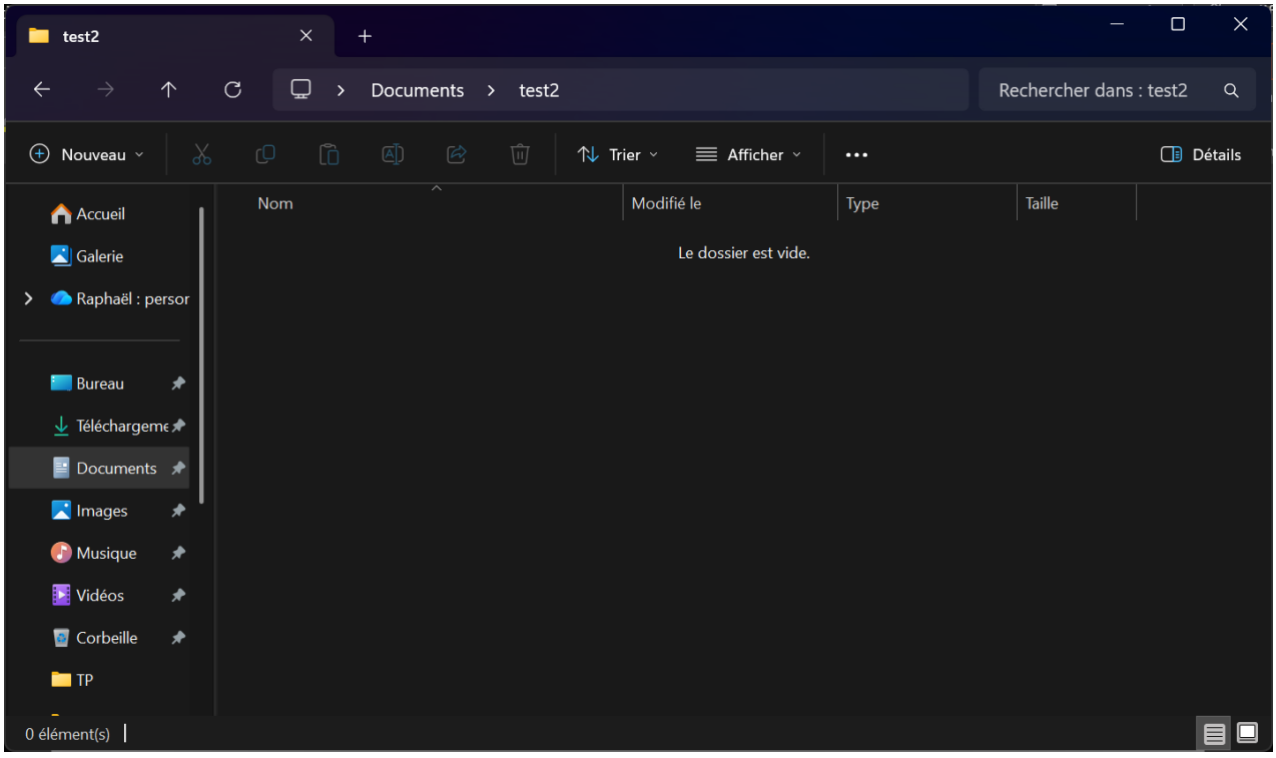
Cette fois ci, nous allons faire une synchronisation locale entre 2 dossiers :

`C:\Users\rdubois\Documents\test1`

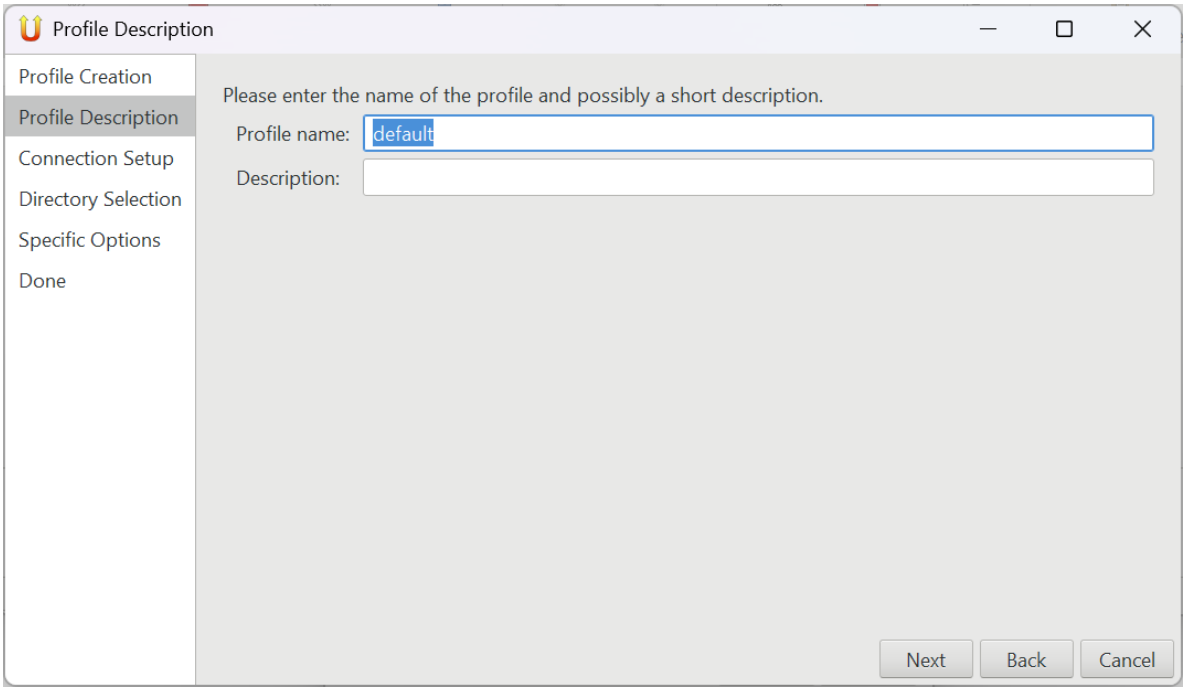
`C:\Users\rdubois\Documents\test2`

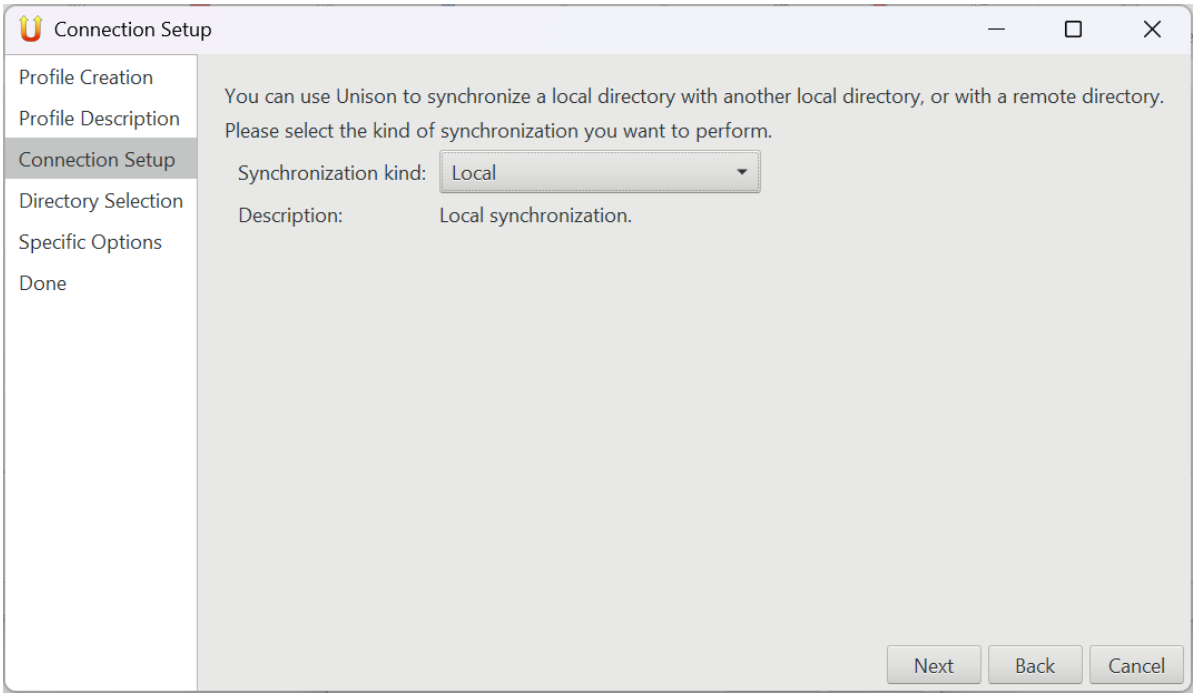
Le dossier test1 contenant un fichier test3.txt :



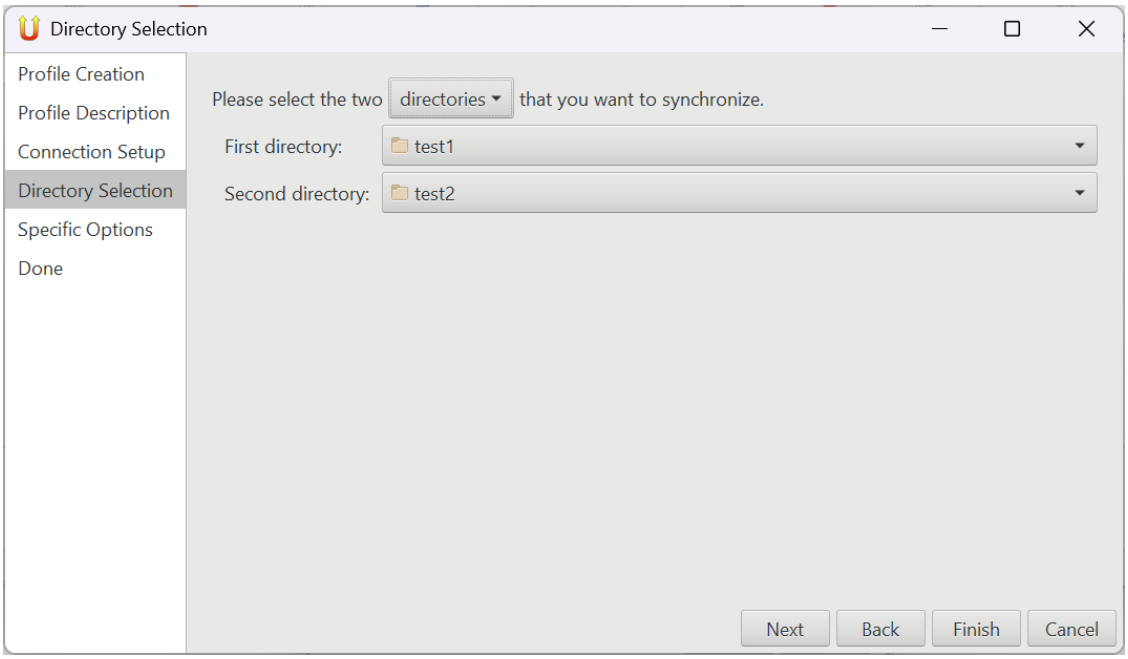


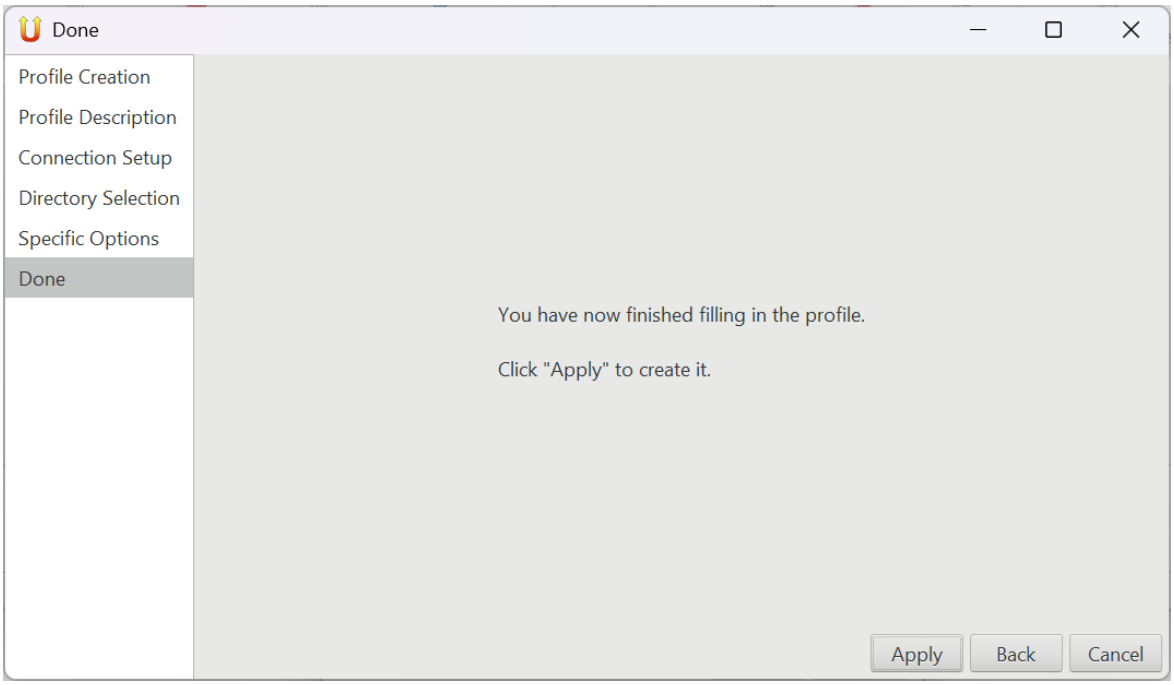
La configuration de Unison est similaire à celle sur Linux, cependant, cette fois ci, nous allons faire une synchronisation Locale et non SSH :



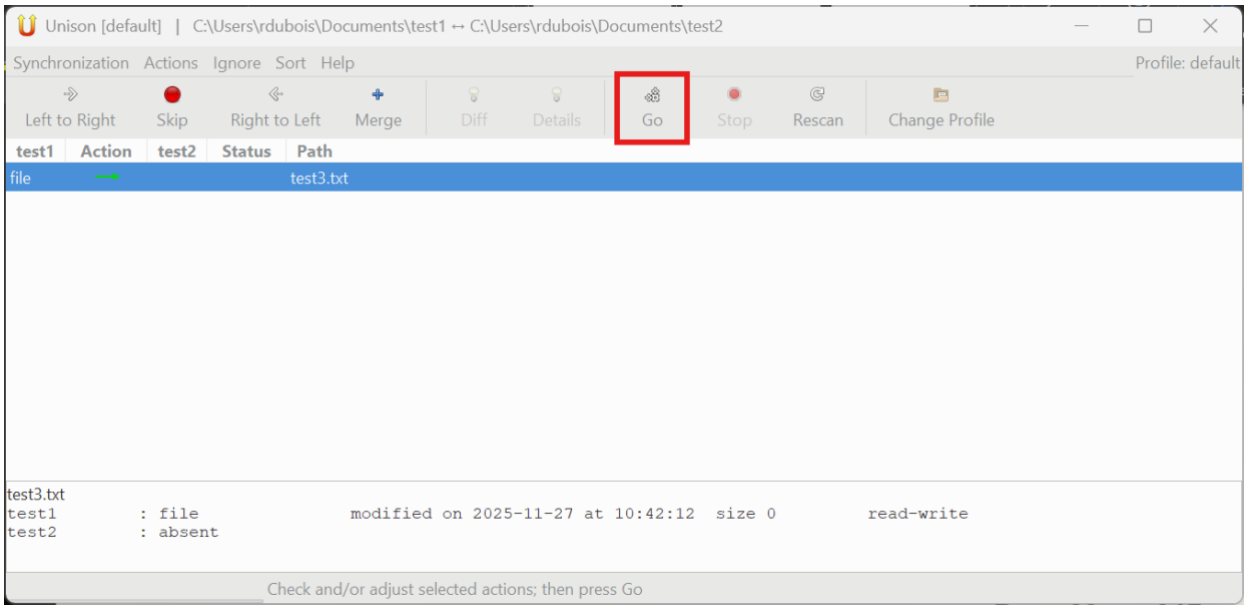


Nous allons ensuite sélectionner les deux dossiers à synchroniser :

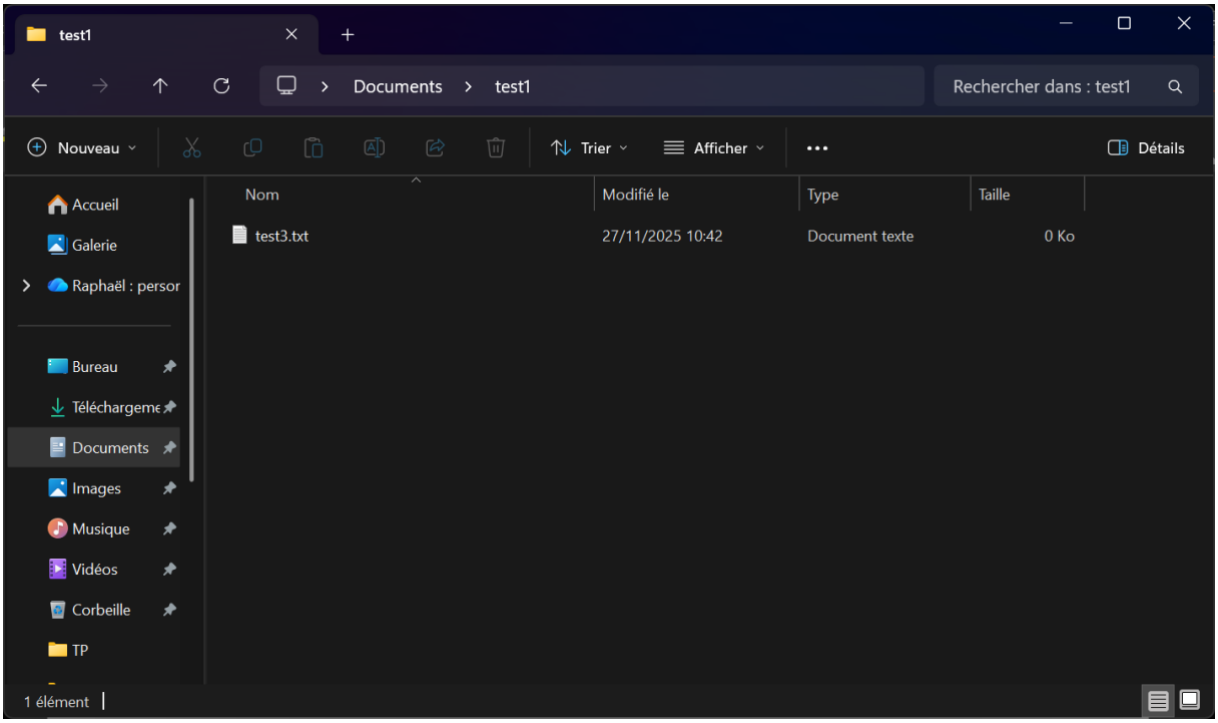


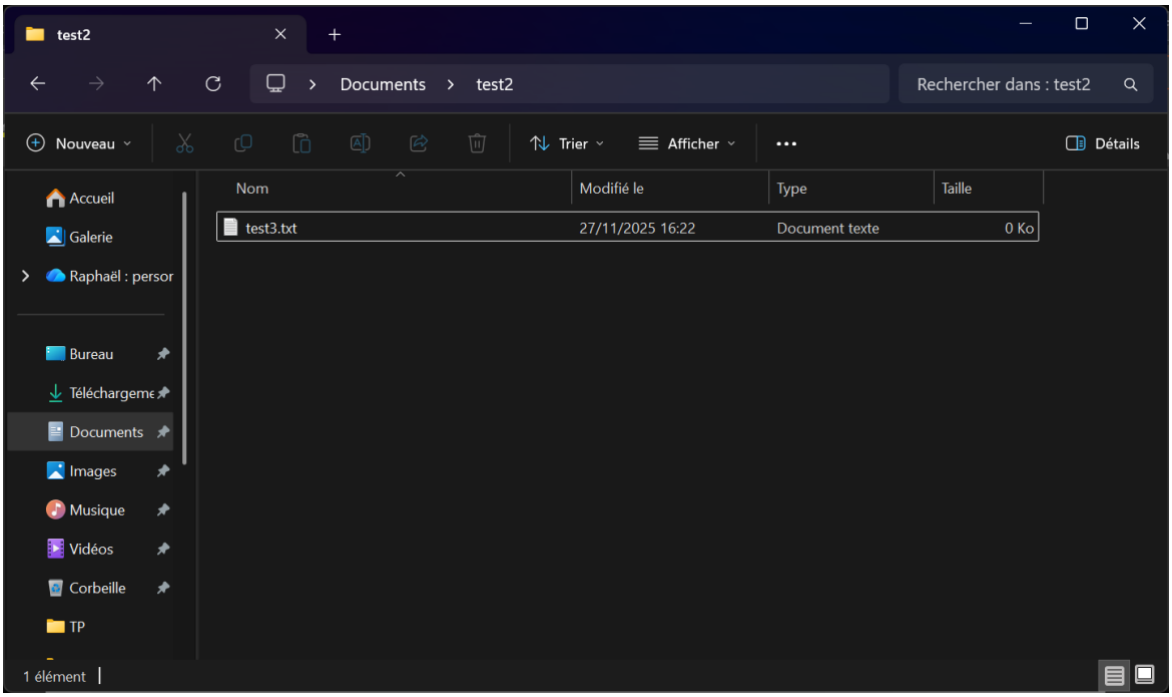


Une fois la configuration terminée, le fichier text3.txt est prêt pour la synchronisation, il suffit de cliquer sur « go » pour lancer cette dernière :



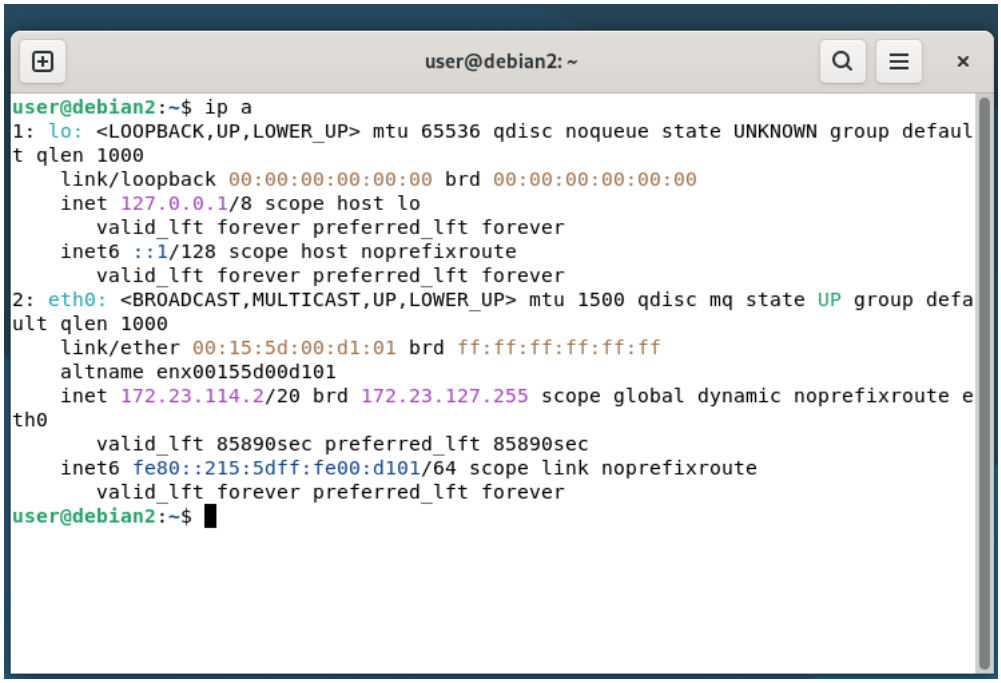
Une fois que nous avons cliqué dessus, on peut vérifier le contenu de test1 et test2 pour vérifier qu'ils contiennent bien tout deux le fichier test3.txt :



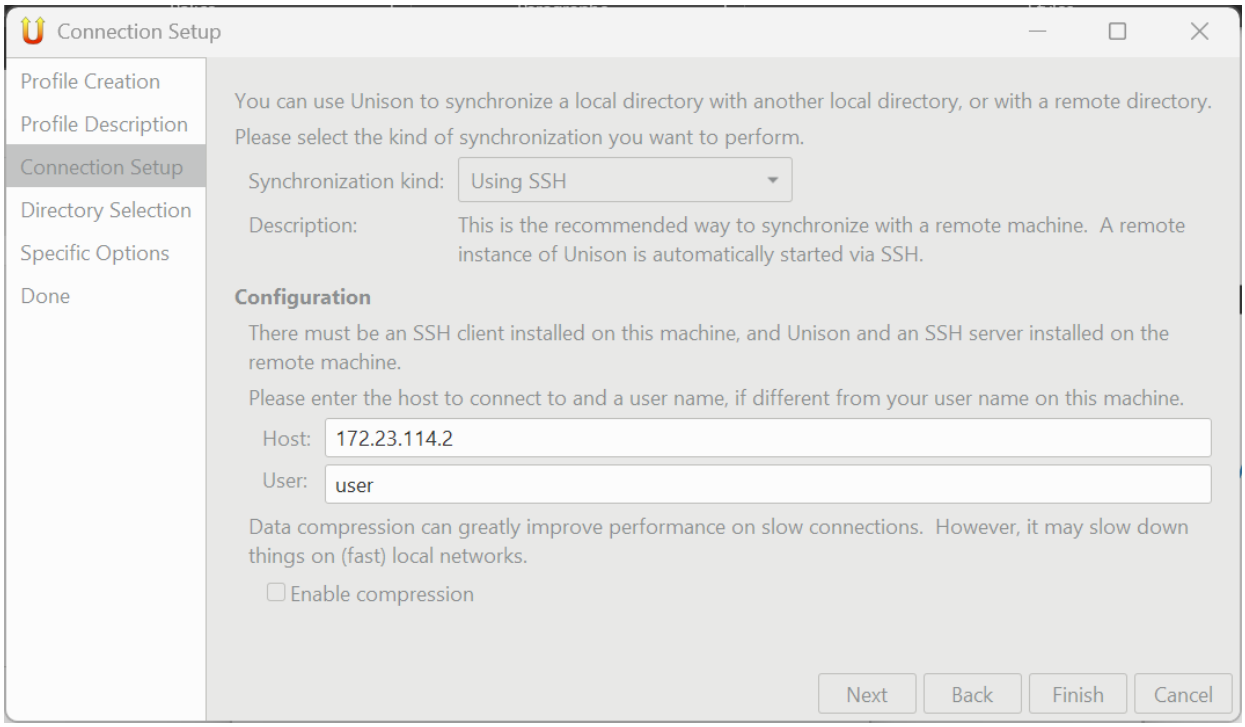


3) Windows / Linux

Cette fois ci, nous allons synchroniser notre VM Linux avec l'hôte Windows, pour cela nous auront besoin de l'adresse IP de la VM (avec Unison et OpenSSH déjà installé) :

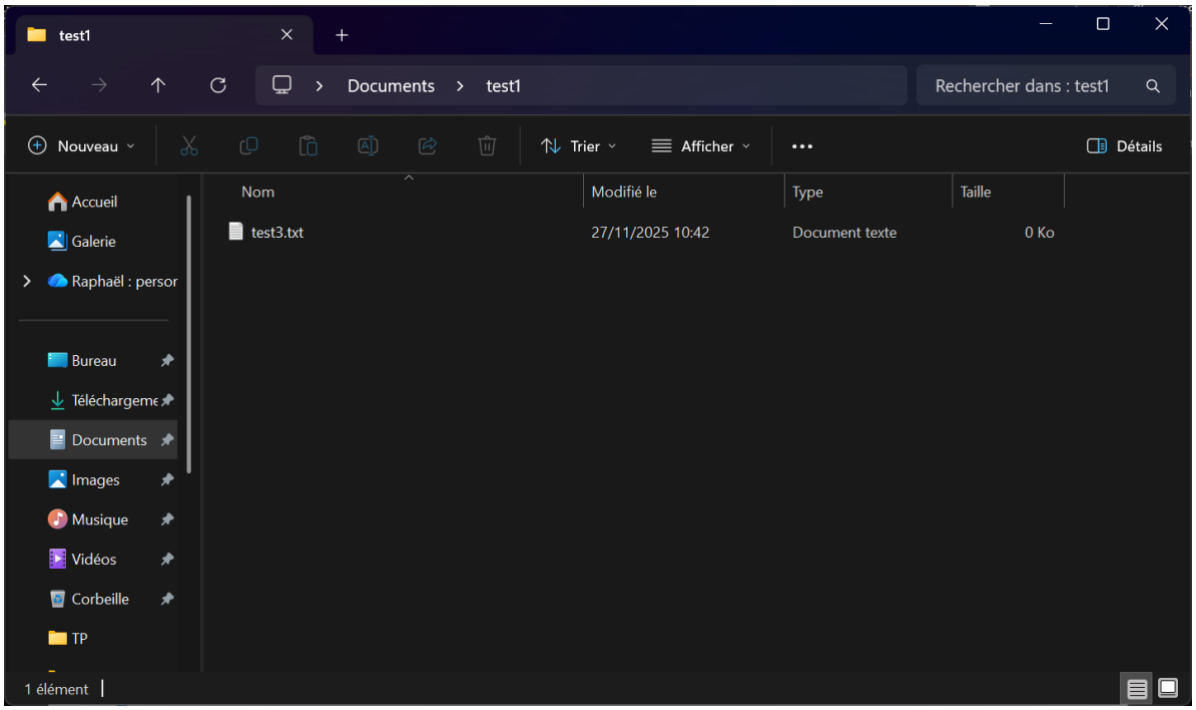


Une fois que nous avons ceci, nous pouvons créer un nouveau profil sur Unison Windows en SSH vers l'adresse IP de notre VM :

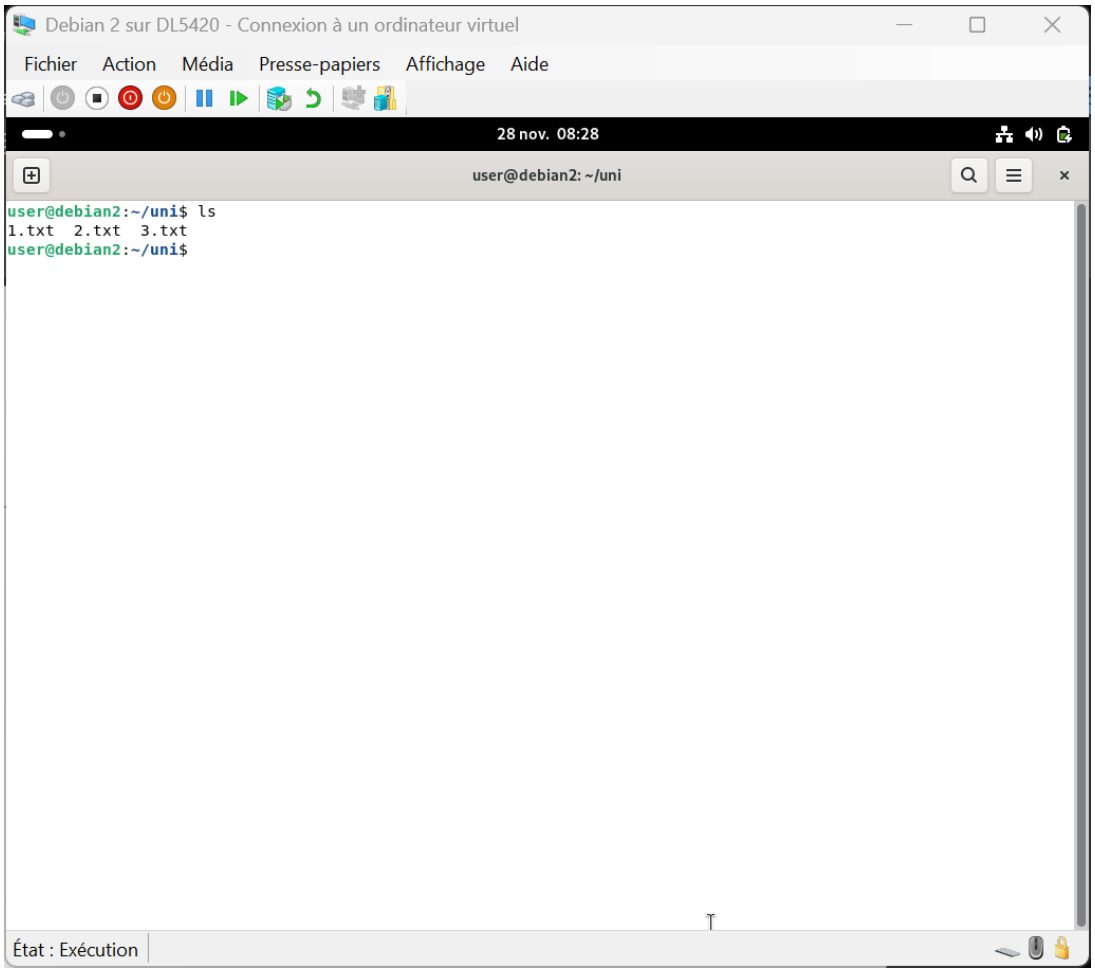


Nous allons aussi définir les dossiers à synchroniser :

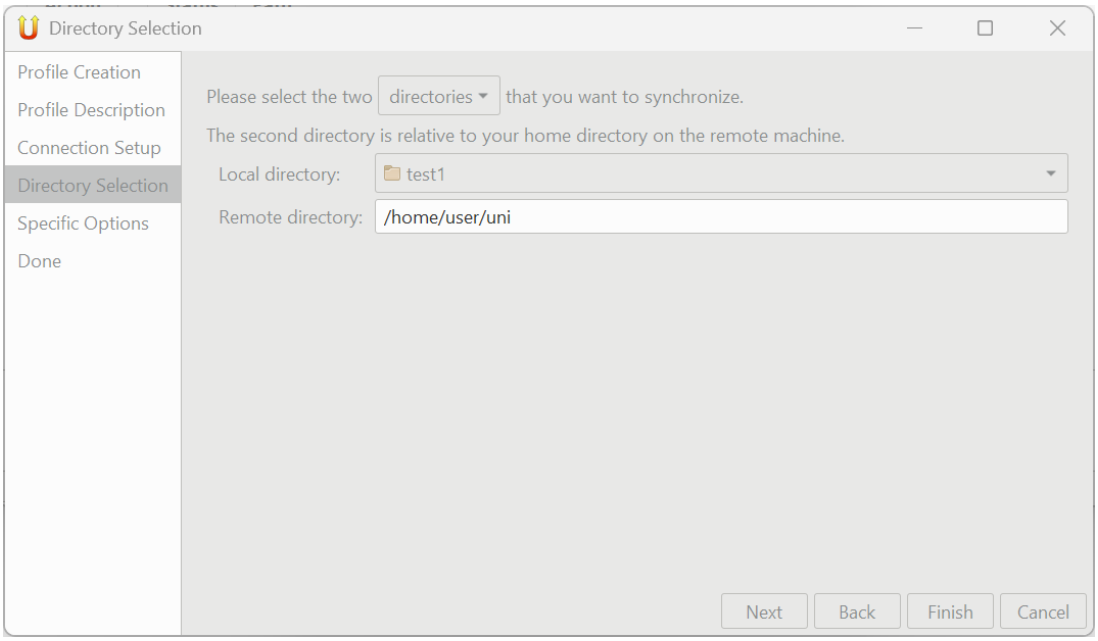
```
Windows : C:\Users\rdubois\Documents\test1
```



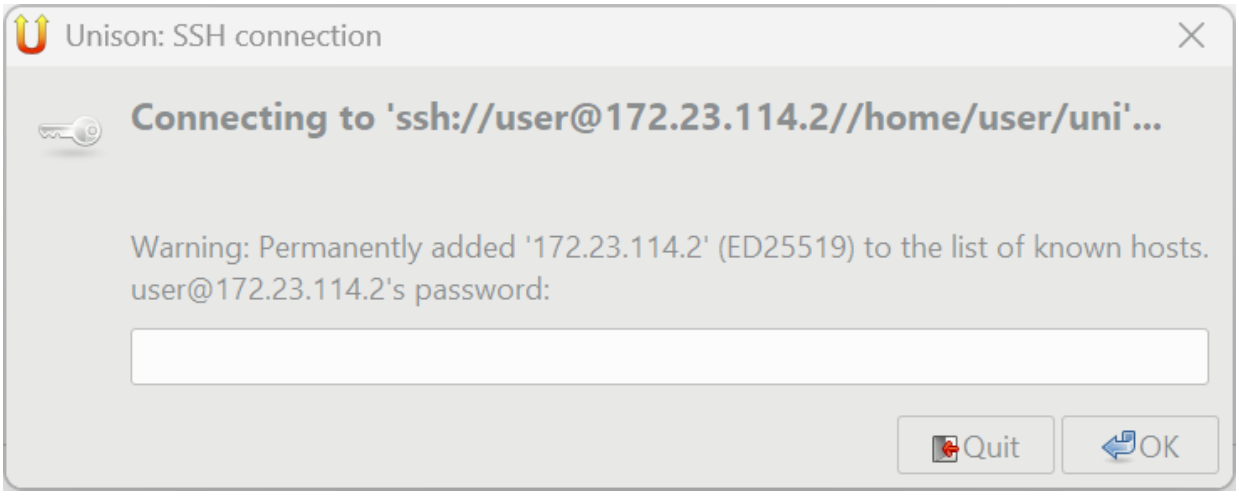
Linux : /home/user/uni



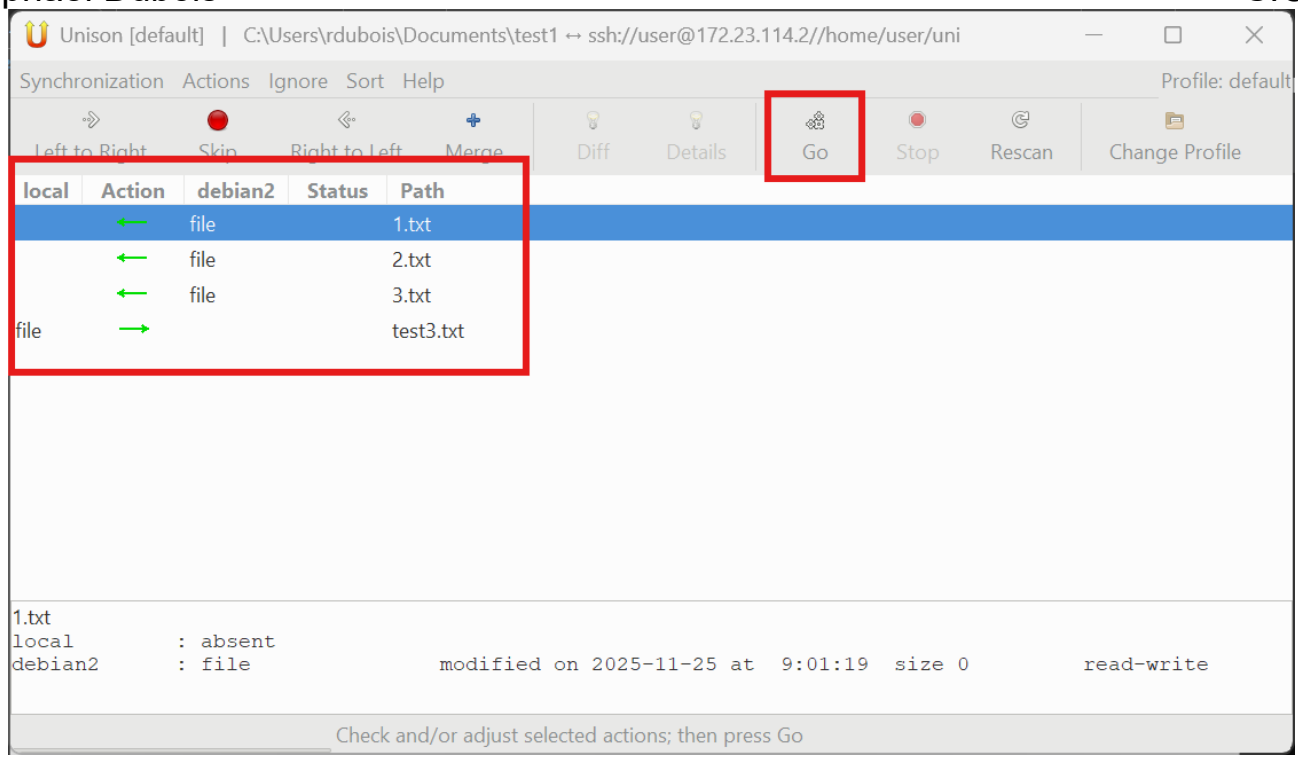
Paramétrage dans Unison :



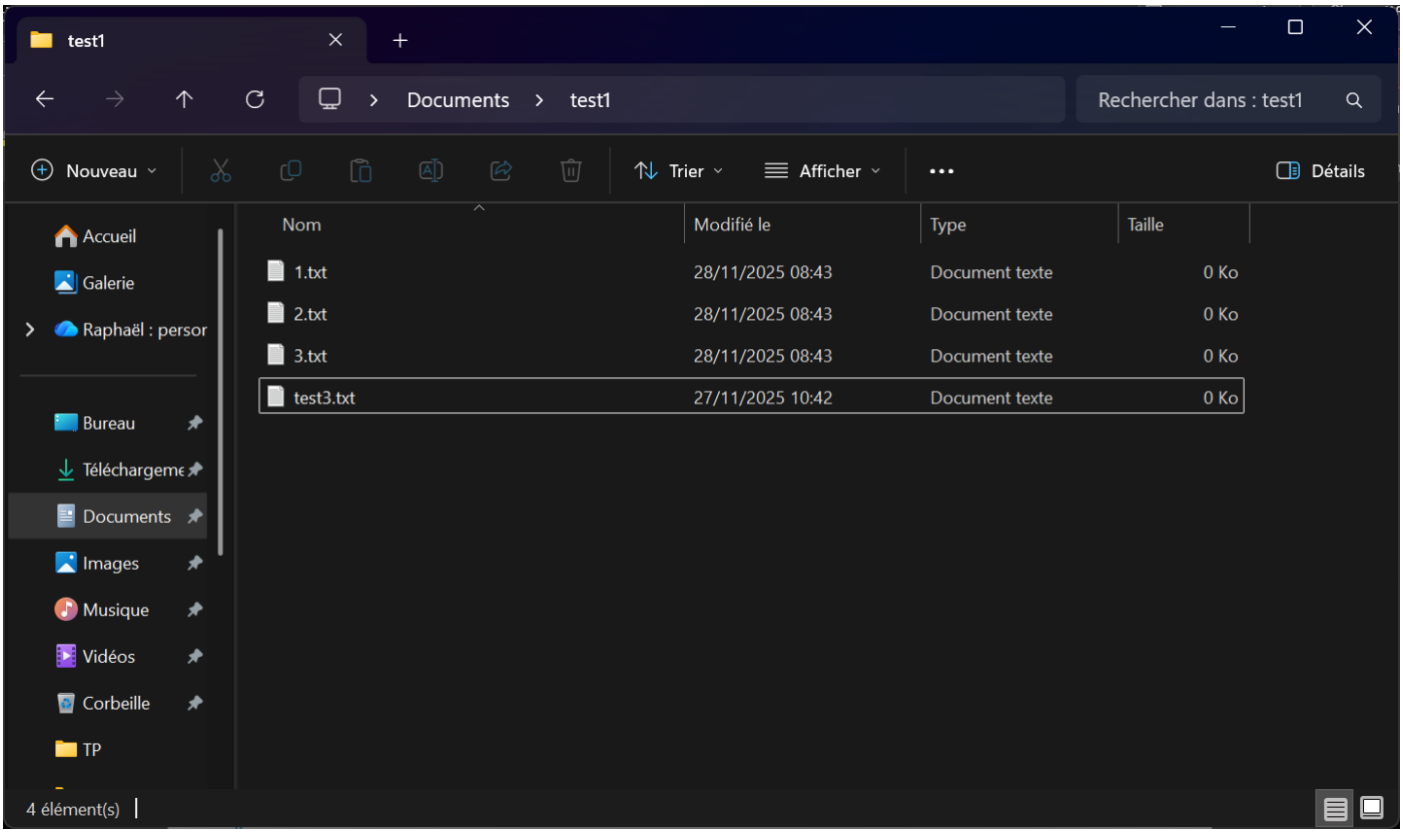
Il nous suffit à présent de cliquer sur Finish, Apply, d'ouvrir notre profil et d'insérer notre mot de passe SSH :



Une fois connecté, nous avons à la fois les fichiers présents sur Windows et Linux qui vont être synchroniser, pour démarrer la synchronisation, il suffit de cliquer sur « Go » :



Une fois ceci fait, les fichiers présents sur Windows sont présents sur le serveur Linux, et inversement !



```
user@debian2:~/uni$ ls
1.txt  2.txt  3.txt  test3.txt
user@debian2:~/uni$
```