

A d m i n i s t r a t i o n

S y s t è m e

G N U / L i n u x

Chapitre 5. NFS, SAMBA

NFS :

I. Présentation

Le NFS, *Network File System*, "*Système de fichiers en réseau*", est **un protocole de transferts de fichiers par le réseau principalement utilisé sur les systèmes Linux/Unix.**

L'utilisateur, à partir de son ordinateur, va pouvoir accéder à des fichiers stockés sur un serveur distant, à l'aide du protocole NFS qui fonctionne selon le mode client/serveur.

Contrairement au SMB qui est un protocole propriétaire Microsoft, le protocole NFS est libre.

Il s'installe sur Linux par l'intermédiaire d'un paquet dédié, et il est pris en charge par des solutions comme VMware ESXi pour connecter une banque de données, mais aussi par les NAS présent sur le marché (Synology, Asustor, Qnap, etc.).

Le protocole NFS est majoritairement utilisé pour connecter un espace de stockage entre un serveur NFS et un ou plusieurs serveurs, par exemple pour stocker des sauvegardes ou héberger des machines virtuelles.

Chapitre 5. NFS, SAMBA

NFS :

II. Les différentes versions de NFS

Développé par Sun Microsystems en 1984, le protocole NFS existe depuis plusieurs dizaines d'années.

Les versions 1 et 2 du protocole NFS fonctionnaient à l'aide de connexion UDP, tandis que le TCP a fait son apparition avec le NFS v3.

Les premières versions, que ce soit NFS v1, NFS v2 ou NFS v3 n'étaient pas sécurisées : à l'époque, la sécurité n'était pas une priorité.

Il y avait également des limitations sur la taille des paquets (8 Ko) et la taille maximale d'un fichier transférable (2 Go) avant l'arrivée de NFS v3.

C'est en **2000**, avec la sortie de **NFS v4** que le protocole a fortement évolué pour intégrer **des fonctionnalités liées à la sécurité**.

Chapitre 5. NFS, SAMBA

NFS :

Le protocole NFS v4 prend en charge complètement la sécurité avec l'authentification via Kerberos et il s'appuie sur des connexions TCP avec suivi de l'état (*stateful*).

Voici un tableau récapitulatif des versions du protocole NFS :

LES VERSIONS DU PROTOCOLE NFS	
VERSION	ANNÉE DE SORTIE / RFC
NFS V1	1984
NFS V2	1989 / RFC 1094
NFS V3	1995 / RFC 1813
NFS V4	2000 ET 2003 / RFC 3010 PUIS RFC 3530
NFS V4.1	2010 / RFC 5661
NFS V4.2	2016 / RFC 7862

Chapitre 5. NFS, SAMBA

NFS :

III. Compatibilité entre les versions de NFS

Il n'y a pas de rétrocompatibilité entre le NFS v4 et les versions précédentes, y compris avec le NFS v3.

Les différences sont trop nombreuses, tant sur l'aspect de la sécurité, mais aussi le fonctionnement des connexions : *TCP stateful* avec NFS v4.

Autrement dit, si le serveur fonctionne uniquement avec NFS v4, le client ne pourra pas utiliser NFS v3.

Les deux doivent effectuer la connexion en NFS v3.

Généralement, les serveurs NFS sont capables de gérer plusieurs versions.

Par contre, si vous souhaitez utiliser l'authentification Kerberos, ce sera obligatoirement en NFS v4, seule version à prendre en charge cette sécurité.

IV. Quel est le port utilisé par le protocole NFS ?

Le protocole NFS s'appuie sur un seul port pour fonctionner : le port 2049.

Il s'agit du **port d'écoute d'un serveur NFS**.

C'est à **partir de NFS v4 que le protocole utilise un seul port**. Les anciennes versions utilisaient le **port 2049**, mais aussi le **port 111** correspondant au service "*portmap*" (utile pour orienter les requêtes RPC vers le bon service).

Chapitre 5. NFS, SAMBA

NFS :

V. Créer un partage NFS sous Debian (Linux)

Pratique:

Nous allons utiliser deux machines sous Linux (Debian), mais vous pouvez utiliser d'autres distributions, l'Ubuntu est similaire car elle découle de la Debian

- **Un serveur Debian** qui jouera le rôle de **serveur NFS**, avec un partage : `/srv/partagenfs`
 - Adresse IP : 192.168.100.121/24
- **Un poste client Debian** qui jouera le rôle de **client NFS**
 - Adresse IP : 192.168.100.120/24

Préparer le serveur. Ensuite, monter le partage NFS sur le poste client et finir par une analyse de trames avec *Tcpdump*.

Note : pas l'authentification Kerberos car cela complexifie la procédure.

Chapitre 5. NFS, SAMBA

NFS :

A. Installation du paquet

Sur le serveur Debian, mettez à jour le cache des paquets :

```
sudo apt-get update
```

Ensuite, installez le paquet "*nfs-kernel-server*" :

```
sudo apt-get install nfs-kernel-server
```

```
root@debian-11:~# apt-get install nfs-kernel-server
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 :
  keyutils libevent-2.1-7 libnfsidmap2 nfs-common rpcbind
Paquets suggérés :
  open-iscsi watchdog
Les NOUVEAUX paquets suivants seront installés :
  keyutils libevent-2.1-7 libnfsidmap2 nfs-common nfs-kernel-server rpcbind
0 mis à jour, 6 nouvellement installés, 0 à enlever et 23 non mis à jour.
Il est nécessaire de prendre 682 ko dans les archives.
Après cette opération, 20028 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] O
Réception de :1 http://ftp.fr.debian.org/debian bullseye/main amd64 rpcbind amd64 1.2.5-9 [51,4 kB]
Réception de :2 http://ftp.fr.debian.org/debian bullseye/main amd64 keyutils amd64 1.6.1-2 [52,8 kB]
Réception de :3 http://ftp.fr.debian.org/debian bullseye/main amd64 libevent-2.1-7 amd64 2.1.12-stab
Réception de :4 http://ftp.fr.debian.org/debian bullseye/main amd64 libnfsidmap2 amd64 0.25-6 [32,6
Réception de :5 http://ftp.fr.debian.org/debian bullseye/main amd64 nfs-common amd64 1:1.3.4-6 [232
Réception de :6 http://ftp.fr.debian.org/debian bullseye/main amd64 nfs-kernel-server amd64 1:1.3.4-
```

Installation serveur NFS sous Debian

Chapitre 5. NFS, SAMBA

NFS :

Nous allons configurer le serveur NFS pour qu'il démarre automatiquement avec le système :

```
sudo systemctl enable nfs-server.service
```

B. Déclarer un partage NFS `/etc/exports`

Commençons par créer le partage :

```
mkdir /srv/partagenfs
```

Puis, on applique les droits sur le partage (à adapter selon vos besoins) :

```
chown nobody:nogroup /srv/partagenfs/ chmod 755 /srv/partagenfs/
```

Pour déclarer les partages NFS, il faut s'appuyer sur le fichier `"/etc/exports"`. C'est dans ce fichier que nous allons déclarer notre fichier `"/srv/partagenfs"`. Editez le fichier :

```
sudo nano /etc/exports
```

Dans ce fichier, voici la ligne à ajouter pour déclarer notre partage :

```
/srv/partagenfs 192.168.100.0/24(rw,sync,anonuid=65534,anongid=65534,no_subtree_check)
```


Chapitre 5. NFS, SAMBA

NFS :

```
GNU nano 5.4
# /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)
#
/srv/partagenfs 192.168.100.0/24(rw,sync,anonuid=65534,anongid=65534,no_subtree_check)
```

- **/srv/partagenfs** : le chemin local du dossier à partager
- **192.168.100.0/24** : l'adresse IP ou le réseau à autoriser, si vous souhaitez autoriser seulement une adresse IP spécifique, précisez cette adresse IP
- **()** : les options pour le partage
- **rw** : partage accessible en lecture et écriture, à remplacer par "**ro**" pour la lecture seule
- **sync** : écrire les données et les vérifier avant de répondre à la requête suivante : plus lent, mais plus fiable vis-à-vis des corruptions de données. L'autre mode est "**async**".
- **anonuid** : ID de l'utilisateur à utiliser pour les connexions anonymes (*65534 = nobody*)
- **anongid** : ID du groupe à utiliser pour les connexions anonymes (*65534 = nogroup*)
- **no_subtree_check** : désactiver la vérification des sous-dossiers, recommandé pour des raisons de fiabilité

Chapitre 5. NFS, SAMBA

NFS :

Il est à noter que l'on peut déclarer plusieurs adresses IP ou réseaux avec des autorisations différentes (ou identiques). Par exemple :

```
/srv/partagenfs 192.168.100.0/24(rw,sync,anonuid=65534,anongid=65534,no_subtree_check) 10.10.10.0/24(rw,sync,anonuid=65534,anongid=65534,no_subtree_check)
```

En complément, vous pouvez lire la documentation : [nfs - man page](#)
Enregistrez le fichier et fermez-le. Pour que la configuration soit prise en compte, il faut utiliser *exportfs*, comme ceci :

```
exportfs -a
```

Il est à noter que pour **stopper et purger les partages NFS**, il faut exécuter la commande suivante (à faire avant la commande précédente pour actualiser) :

```
exportfs -ua
```

La commande ci-dessous permet d'afficher la liste des partages NFS sur l'hôte précisé, en l'occurrence notre machine Debian elle-même.

```
showmount -e 192.168.100.121
```

Chapitre 5. NFS, SAMBA

NFS :

```
root@debian-11:~# showmount -e 192.168.100.121
Export list for 192.168.100.121:
/srv/partagenfs 10.10.10.0/24,192.168.100.0/24
```

Exemple de la commande showmount nfs

On peut voir le nom du partage *"/srv/partagenfs"* ainsi que les deux sous-réseaux autorisés

La commande ci-dessous est à connaître également pour lister les services en écoute via RPC.

On peut voir le service "nfs" avec différentes versions, notamment NFS v3 et NFS v4.

On voit également la présence de *portmapper* sur le port 111 (service non spécifique à NFS, mais utilisé par certaines versions).

```
rpcinfo -p
```

Chapitre 5. NFS, SAMBA

NFS :

```
root@debian-11:~# rpcinfo -p
program vers proto  port  service
100000    4    tcp    111    portmapper
100000    3    tcp    111    portmapper
100000    2    tcp    111    portmapper
100000    4    udp    111    portmapper
100000    3    udp    111    portmapper
100000    2    udp    111    portmapper
100005    1    udp    41278  mountd
100005    1    tcp    34225  mountd
100005    2    udp    36289  mountd
100005    2    tcp    32791  mountd
100005    3    udp    39225  mountd
100005    3    tcp    44325  mountd
100003    3    tcp    2049   nfs
100003    4    tcp    2049   nfs
100227    3    tcp    2049
100003    3    udp    2049   nfs
100227    3    udp    2049
100021    1    udp    37581  nlockmgr
100021    3    udp    37581  nlockmgr
100021    4    udp    37581  nlockmgr
100021    1    tcp    38809  nlockmgr
100021    3    tcp    38809  nlockmgr
100021    4    tcp    38809  nlockmgr
```

Aperçu de NFS avec rpcinfo

Chapitre 5. NFS, SAMBA

NFS :

Sur le poste client:

C. Connexion au partage NFS depuis Linux

Sur le poste Debian, mettez à jour le cache des paquets :

```
sudo apt-get update
```

Puis, installez le paquet "*nfs-common*" afin de pouvoir monter le partage NFS sur l'hôte local.

```
sudo apt-get install nfs-common
```

```
$ sudo apt-get install nfs-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  keyutils libevent-2.1-7 libmpdec3 libnfsidmap2 libpython3-stdlib libpython3.9-minimal libpython3.9
  libreadline8 libsqlite3-0 libwrap0 media-types python3 python3-minimal python3.9 python3.9-minimal
Suggested packages:
  open-iscsi watchdog python3-doc python3-tk python3-venv python3.9-venv python3.9-doc binfmt-support
The following NEW packages will be installed:
  keyutils libevent-2.1-7 libmpdec3 libnfsidmap2 libpython3-stdlib libpython3.9-minimal libpython3.9
  libreadline8 libsqlite3-0 libwrap0 media-types nfs-common python3 python3-minimal python3.9 python
  rpcbind
0 upgraded, 17 newly installed, 0 to remove and 20 not upgraded.
Need to get 6,702 kB of archives.
After this operation, 23.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

Installation d'un client NFS sous Debian

Chiheb G.

Chapitre 5. NFS, SAMBA

NFS :

Maintenant, nous allons pouvoir monter manuellement le partage NFS sur notre machine. Pour monter le partage, nous devons créer un dossier local qui servira de point de montage. Créez ce dossier :

```
mkdir /media/partagenfs
```

Ensuite, la commande "mount" va nous permettre de monter notre partage NFS. Ce qui donne :

```
mount -t nfs4 192.168.100.121:/srv/partagenfs /media/partagenfs/
```

Explications :

- **-t nfs4** : utiliser NFS v4 pour monter ce partage.
- **192.168.100.121:/srv/partagenfs** : partage NFS à monter, sous la forme *<hote>:<chemin-partage>*
- **/media/partagenfs/** : point de montage local

À la suite de cette commande, créer un fichier sur le partage :

```
touch /media/partagenfs/monfichier.txt
```

Chapitre 5. NFS, SAMBA

NFS :

Sur le serveur NFS, on peut voir les différents fichiers créés :

```
root@debian-11:~# ls -l /srv/partagenfs/
total 0
-rw-r--r-- 1 nobody nogroup 0 11 oct. 13:59 monfichier2.txt
-rw-r--r-- 1 nobody nogroup 0 11 oct. 14:18 monfichier3.txt
-rw-r--r-- 1 nobody nogroup 0 11 oct. 13:59 monfichier.txt
```

Il faut savoir qu'avec la commande "mount", le montage sera temporaire, c'est-à-dire que si l'on redémarrer le poste client, le partage ne sera pas monté automatiquement.

Démontez le partage et nous allons le monter différemment :

```
umount /media/partagenfs
```

Ensuite, modifiez la table de montage "[/etc/fstab](#)" :

```
sudo nano /etc/fstab
```

Spécifiez la ligne suivante pour le monter automatiquement, toujours via NFS v4 :

```
192.168.100.121:/srv/partagenfs /media/partagenfs nfs4 defaults,user,exec 0 0
```

Chapitre 5. NFS, SAMBA

NFS :

Enregistrez puis exécutez la commande ci-dessous pour charger le contenu du fichier `" /etc/fstab"` et monter notre partage.

```
sudo mount -a
```

Voilà, vous devriez pouvoir accéder au partage NFS de la même manière qu'en le montant avec *mount*.

Chapitre 5. NFS, SAMBA

SAMBA :

Samba est une suite logicielle open source fournissant **des services de partage de fichiers et d'impression** entre les systèmes **Windows** et **Linux** .

Sa popularité croissante est due à son **interopérabilité** et à sa **polyvalence** pour **intégrer** Linux et Windows.

Outre le partage de fichiers, Samba prend en charge des fonctionnalités telles que l'authentification et le contrôle d'accès, s'intégrant aux services de stockage cloud et au streaming de fichiers multimédias.

Samba s'appuie sur le protocole **smb** ou **Server Message Block** , qui est le **protocole de partage réseau principal** pour la communication interprocessus.

Samba est un choix populaire pour la communication sur un réseau en raison de sa **compatibilité multiplateforme** , **de sa gestion des ressources** et **de ses services de partage rapide** .

Pour bénéficier de ces avantages, la tâche principale consiste à installer Samba et à le configurer en fonction de vos besoins.

Chapitre 5. NFS, SAMBA

SAMBA :

Processus étape par étape pour installer et configurer le serveur Samba

Étape 1 : Installer le serveur Samba sur Ubuntu

```
sudo apt update
```

```
ayesha@ubuntu:~$ sudo apt update
[sudo] password for ayesha:
Hit:1 http://bd.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:3 http://bd.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://bd.archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
Get:5 http://bd.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [579 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/main amd64 DEP-11 Metadata [41.6 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 DEP-11 Metadata [18.5 kB]
Get:8 http://bd.archive.ubuntu.com/ubuntu jammy-updates/main amd64 DEP-11 Metadata [102 kB]
Get:9 http://bd.archive.ubuntu.com/ubuntu jammy-updates/universe i386 Packages [609 kB]
Get:10 http://bd.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [887 kB]
Get:11 http://bd.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 DEP-11 Metadata [269 kB]
Get:12 http://bd.archive.ubuntu.com/ubuntu jammy-updates/multiverse i386 Packages [3,884 B]
Get:13 http://bd.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [35.3 kB]
Get:14 http://bd.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 DEP-11 Metadata [940 B]
Get:15 http://bd.archive.ubuntu.com/ubuntu jammy-backports/main amd64 DEP-11 Metadata [7,976 B]
Get:16 http://bd.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 DEP-11 Metadata [12.9 kB]
Fetched 2,903 kB in 7s (391 kB/s)
```

Chapitre 5. NFS, SAMBA

SAMBA :

Installez Samba avec la commande suivante.

```
sudo apt-get install samba -y
```

```
ayesha@ubuntu:~$ sudo apt install samba -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
samba is already the newest version (2:4.15.13+dfsg-0ubuntu1.1).
0 upgraded, 0 newly installed, 0 to remove and 108 not upgraded.
```

```
whereis samba
```

```
ayesha@ubuntu:~$
ayesha@ubuntu:~$ whereis samba
samba: /usr/sbin/samba /usr/lib/x86_64-linux-gnu/samba /etc/samba /usr/share
/samba /usr/share/man/man7/samba.7.gz /usr/share/man/man8/samba.8.gz
```

Location of Samba installed

Chapitre 5. NFS, SAMBA

SAMBA :

Étape 2 : Créer un répertoire pour partager des fichiers

Créer un répertoire partagé dans le système Linux qui soit facilement accessible depuis Linux et Windows:

1. Tapez la commande suivante pour créer un répertoire pour le partage Samba.

```
mkdir /home/ayesha/Desktop/Sambashare
```

Étape 3 : Ajouter un utilisateur au serveur Samba sur Ubuntu

```
sudo smbpasswd -a ayesha
```

```
ayesha@ubuntu:~$ sudo smbpasswd -a ayesha
New SMB password:
Retype new SMB password:
Added user ayesha.
```

Chapitre 5. NFS, SAMBA

SAMBA :

Étape 4 : configurer les paramètres Samba pour le dossier partagé

```
sudo nano /etc/samba/smb.conf
```

```
ayesha@ubuntu:~$ sudo nano /etc/samba/smb.conf ←  
[sudo] password for ayesha:
```

[Smbashare]

```
path = /home/ayesha/Desktop/Smbashare  
available = yes  
valid users = ayesha  
read only = no  
writeable=yes  
browseable=yes  
public = yes
```


Chapitre 5. NFS, SAMBA

SAMBA :

- **smbashare** : Nom du dossier dont vous souhaitez partager le contenu.
- **path** : Chemin absolu du dossier que vous souhaitez partager.
- **available** : Spécifie si le fichier est disponible pour le partage ou non.
- **utilisateur valide** : Nom d'utilisateur.
- **lecture seule** : Spécifie si le fichier est disponible en lecture ou non.
- **writeable** : Spécifie si le dossier peut être écrit ou non.
- **parcourirable** : Spécifie si le fichier partagé est visible lors de la navigation sur le réseau ou non.
- **public** : Spécifie si le fichier partagé est accessible à tous les utilisateurs ou non.

[Smbashare]

```
path = /home/ayesha/Desktop/Smbashare
available = yes
valid users = ayesha
read only = no
writeable=yes
browseable=yes
public = yes
```

```
sudo service smbd restart
```

```
ayesha@ubuntu:~$ sudo service smbd restart ←
[sudo] password for ayesha:
```

Chapitre 5. NFS, SAMBA

SAMBA :

```
systemctl status smbd
```

```
ayesha@ubuntu:~$ systemctl status smbd
● smbd.service - Samba SMB Daemon
   Loaded: loaded (/lib/systemd/system/smbd.service; enabled;
 vendor pres>
   Active: active (running) since Tue 2023-05-16 15:09:41 +06;
 30min ago
     Docs: man:smbd(8)
           man:samba(7)
           man:smb.conf(5)
   Process: 6759 ExecStartPre=/usr/share/samba/update-apparmor-
samba-profi>
   Main PID: 6763 (smbd)
```

Chapitre 5. NFS, SAMBA

SAMBA :

Partager des fichiers à l'aide du serveur Samba sur Ubuntu

Étape 1 : établir une connexion entre Linux et Windows

Pour aider Linux et Windows à communiquer entre eux, vous devez installer les **packages et protocoles de partage de fichiers** requis :

Exécutez la commande ci-dessous pour installer le **package client et utilitaires** sur Samba.

```
sudo apt install cifs-utils samba client -y
```

EXPLICATION

- **cifs-utils** : Package qui prend en charge le montage du système de fichiers CIFS/SMB. CIFS est le principal protocole de partage de fichiers utilisé par Samba.
- **client samba** : package contenant les outils client nécessaires pour les partages Samba et d'autres opérations liées à Samba.
- **-y** : Répond automatiquement « oui » à tout message de confirmation lors de l'installation.

```
ayesha@ubuntu:~$ sudo apt install cifs-utils samba client -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```


Chapitre 5. NFS, SAMBA

SAMBA :

Pour accéder aux fichiers sous Windows, vous devez obtenir l' **adresse IP** de votre système Linux. Exécutez la commande ci-dessous pour le trouver.

```
hostname -I
```

```
ayesha@ubuntu:~$ hostname -I  
192.168.153.128
```

Étape 2 : créer des fichiers dans Ubuntu et y accéder depuis Windows

Dans cette étape, je vais **enregistrer une image** appelée **Photo.jpeg** sous **Linux** et tenter d' **y accéder depuis Windows** :

1.Sous Linux, accédez au dossier partagé, **Smbashare** en suivant le chemin Fichiers>Accueil>Bureau>Smbashare et enregistrez-y le fichier image.

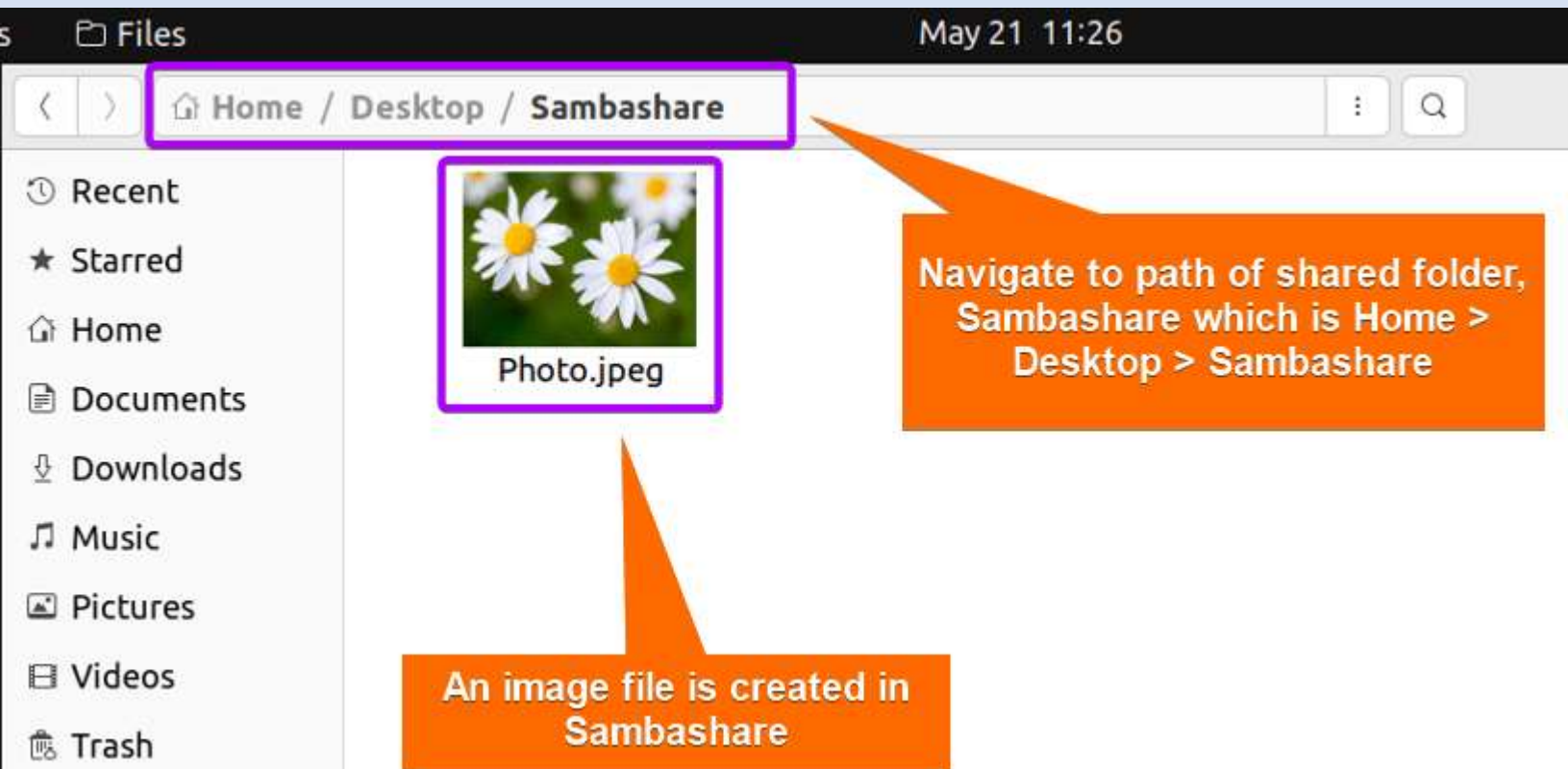
Chapitre 5. NFS, SAMBA

SAMBA :

Étape 2 : créer des fichiers dans Ubuntu et y accéder depuis Windows

Dans cette étape, je vais **enregistrer une image** appelée **Photo.jpeg** sous **Linux** et tenter d' **y accéder depuis Windows** :

1. Sous Linux, accédez au dossier partagé, **Smbashare** en suivant le chemin Fichiers>Accueil>Bureau>Smbashare et enregistrez-y le fichier image.

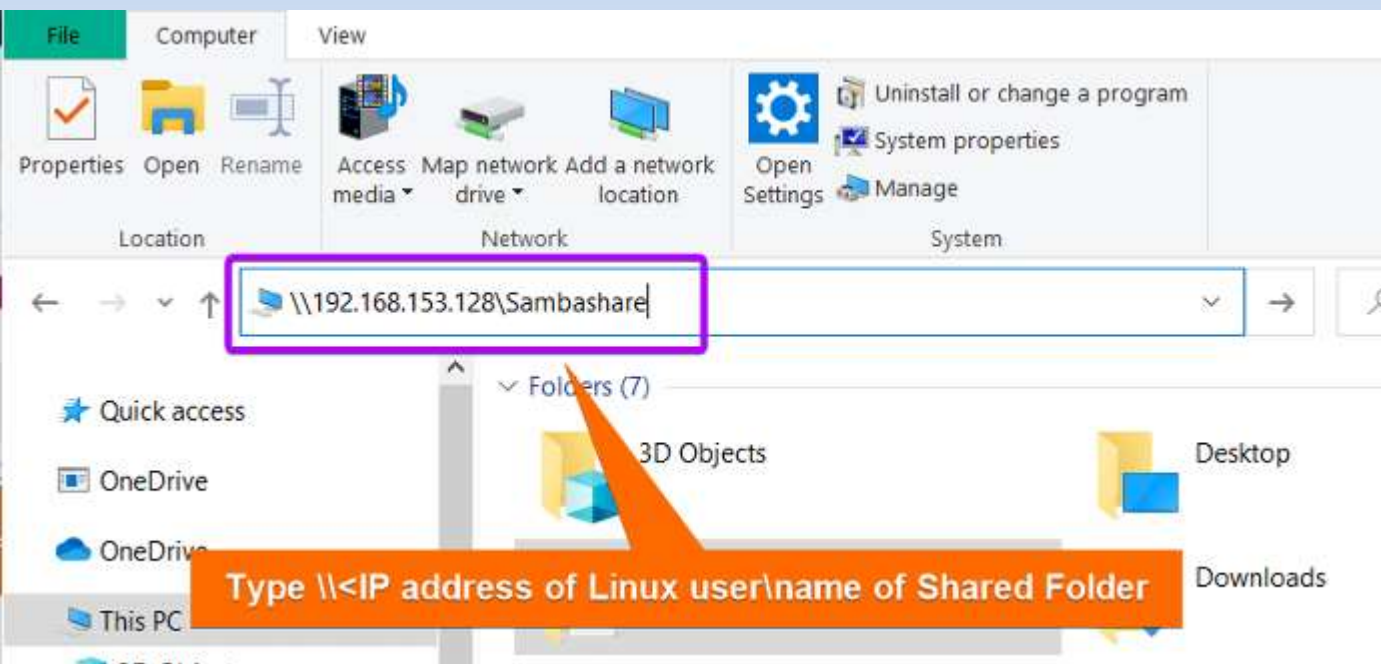


Chapitre 5. NFS, SAMBA

SAMBA :

2. Allez ensuite sous Windows et tapez l'adresse IP du système Linux et le nom du dossier partagé, Sambashare dans la barre d'adresse.

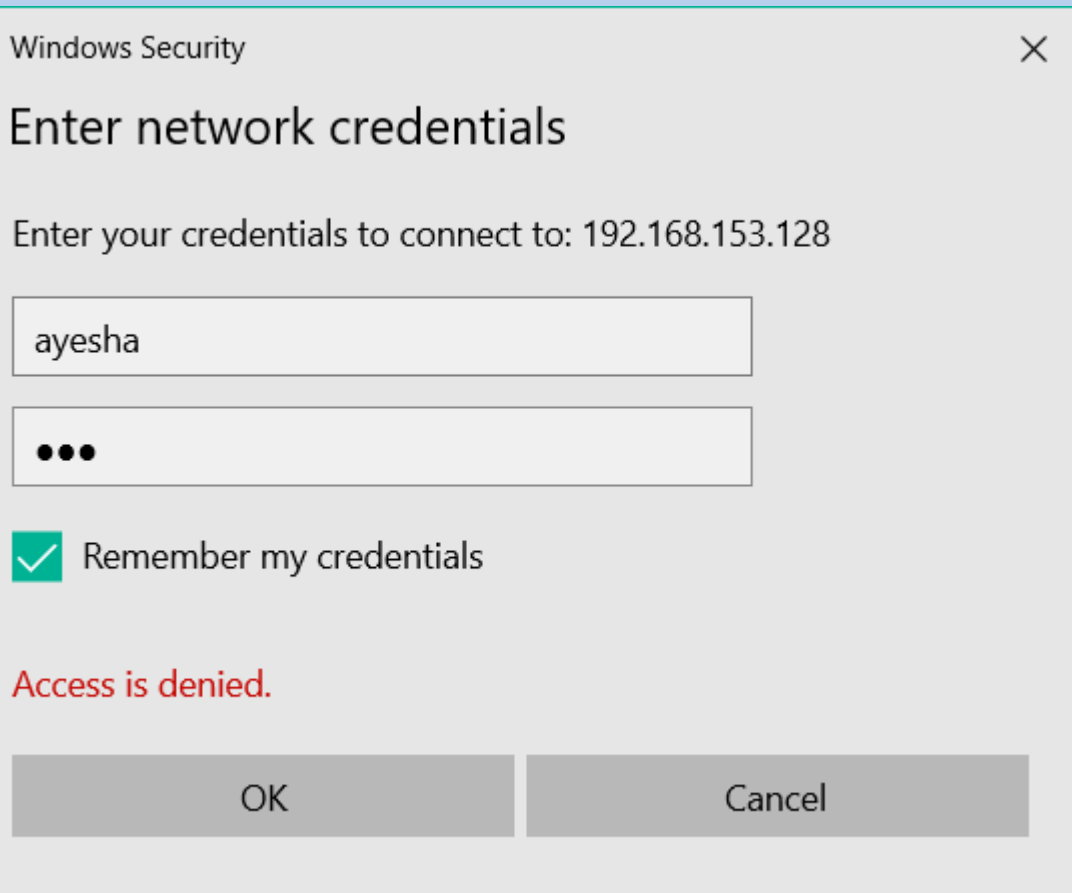
\\192.168.153.128\Sambashare



Chapitre 5. NFS, SAMBA

SAMBA :

3. Une boîte de dialogue comme l'image ci-dessous apparaîtra. Entrez les informations de votre utilisateur Ubuntu actuellement connecté.



A screenshot of a Windows Security dialog box titled "Enter network credentials". The dialog box is light gray with a close button (X) in the top right corner. It prompts the user to "Enter your credentials to connect to: 192.168.153.128". There are two input fields: the first contains the username "ayesha", and the second contains three black dots representing a password. Below the input fields is a checkbox with a green checkmark and the text "Remember my credentials". At the bottom, there is a red error message "Access is denied." and two buttons: "OK" and "Cancel".

Windows Security

Enter network credentials

Enter your credentials to connect to: 192.168.153.128

ayesha

...

☒ Remember my credentials

Access is denied.

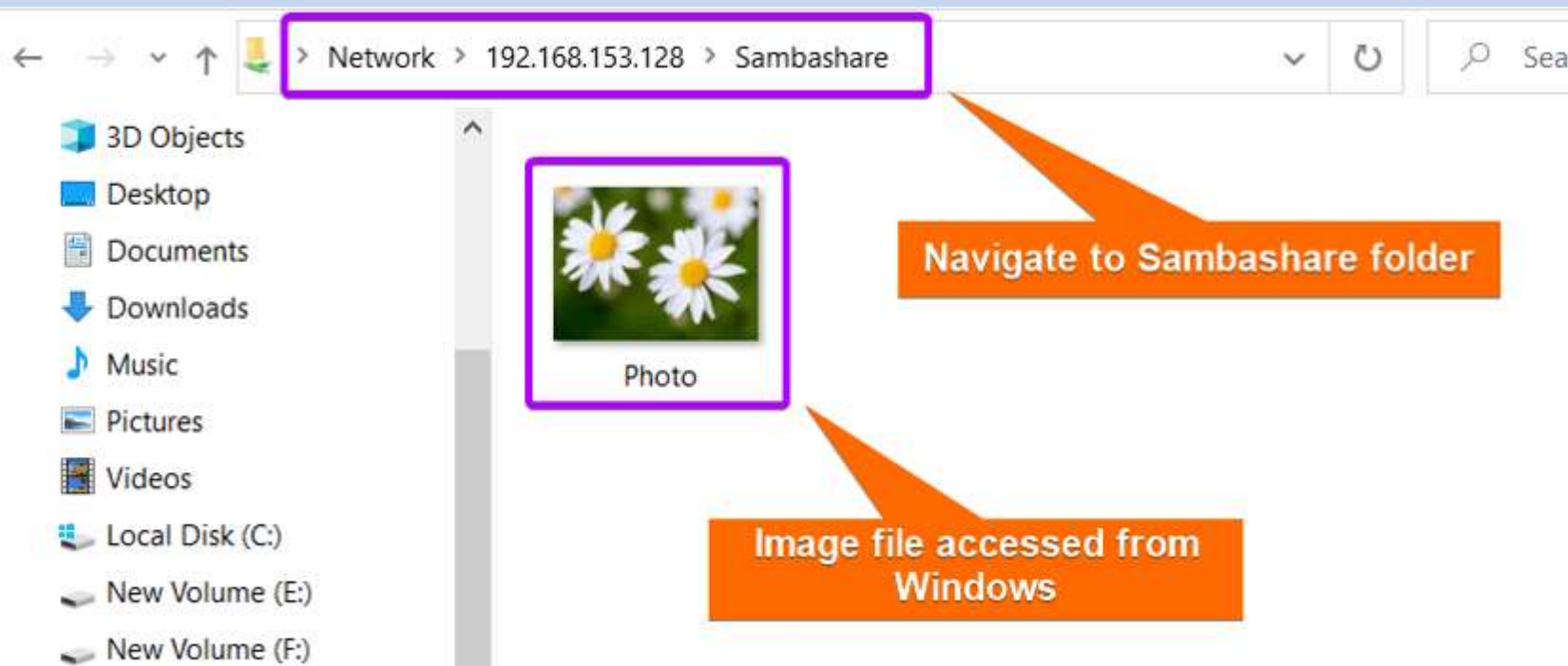
OK Cancel

Chapitre 5. NFS, SAMBA

SAMBA :

1.

4. Vous avez maintenant **accédé avec succès** au dossier **Sambashare** depuis **Windows** et pouvez afficher son contenu.



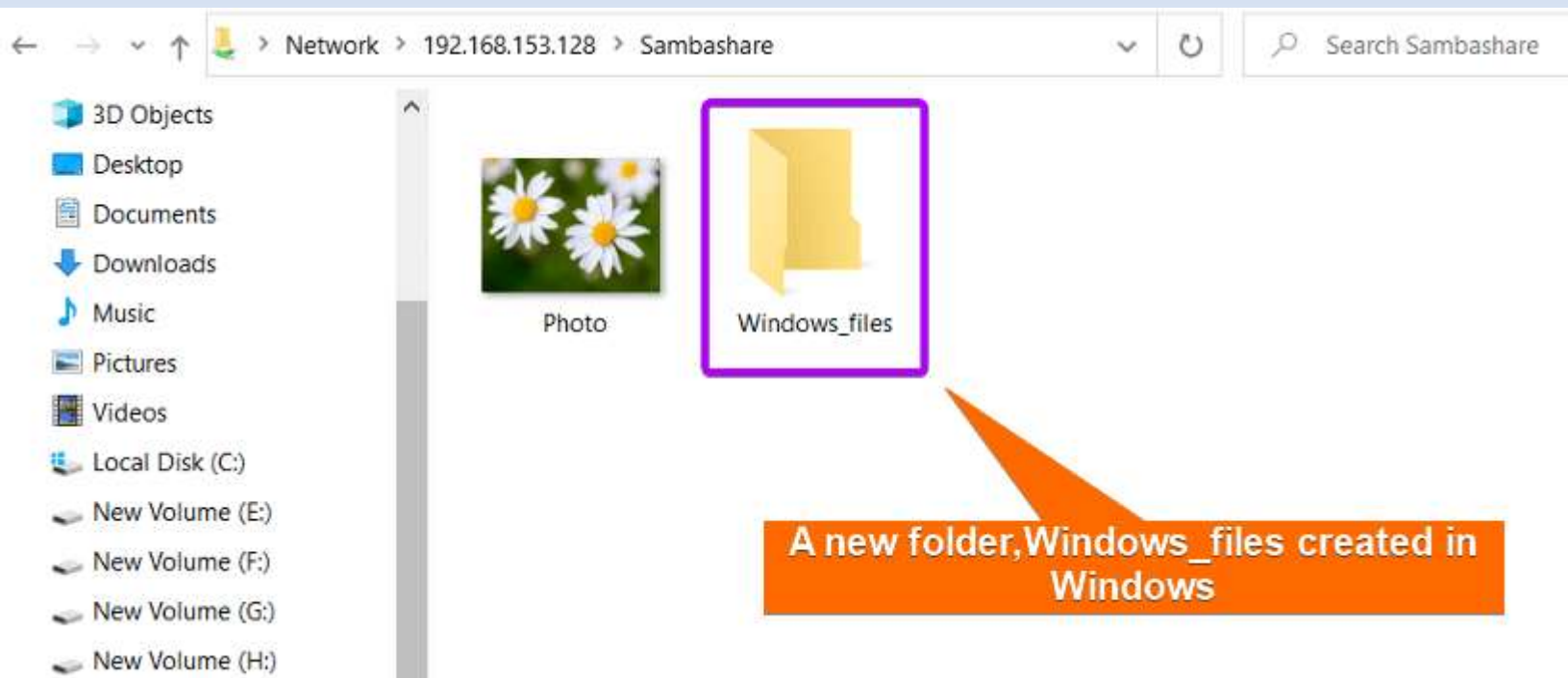
Chapitre 5. NFS, SAMBA

SAMBA :

Étape 3 : créer des fichiers sous Windows et y accéder depuis Ubuntu

Créer un dossier Windows_files dans Sambashare sous Windows et essayer d' y accéder depuis Linux

1.Accédez à Windows et créez un dossier, **Windows_files** .



Chapitre 5. NFS, SAMBA

SAMBA :

2. Ensuite, ouvrez le dossier **Smbashare** sous Linux.

