

Présentation sur NFS

Qu'est-ce que NFS ?

Network File System (NFS), ou système de fichiers en réseau, est une application client/serveur qui permet à un utilisateur de consulter et, éventuellement, de stocker et de mettre à jour des fichiers sur un ordinateur distant, comme s'ils étaient sur son propre ordinateur.

C'est-à-dire, un protocole de partage de fichiers qui permet à plusieurs ordinateurs d'accéder à des fichiers et à des répertoires distants comme s'ils étaient locaux.

Objectifs de NFS

- **Partage centralisé** : Centraliser les fichiers dans un serveur unique accessible à plusieurs clients.
- **Transparence** : Permettre aux utilisateurs d'accéder aux fichiers distants de manière transparente.
- **Efficacité** : Réduire la duplication de fichiers sur plusieurs machines.
- **Compatibilité** : Fonctionner avec différents systèmes d'exploitation comme Linux, UNIX, et même Windows (via des outils spécifiques).

Fonctionnement de NFS

- **Architecture Client-Serveur** :
 - - Un serveur NFS partage des fichiers et des répertoires.
 - - Un client NFS accède aux ressources partagées comme si elles étaient locales.
- **Protocole RPC (Remote Procedure Call)** : NFS repose sur RPC pour permettre la communication entre le client et le serveur.
- **Points de montage** : Les répertoires distants sont montés sur le système de fichiers local pour y accéder facilement.

Versions de NFS

- **NFSv2** : La version initiale, introduite en 1984. Simple mais avec des limitations (taille des fichiers, performances).

NFSv2 est le format le plus ancien et le plus largement pris en charge. Il fonctionne avec le protocole UDP (User Datagram Protocol) via un réseau IP. Le réseau IP permet une connexion réseau stable.

L'UDP de cette configuration ne produit pas formellement une connexion avant de pouvoir commencer à transférer des données. Cette caractéristique est pratique, car elle facilite le fonctionnement des connexions en moins de temps. Mais les clients UDP peuvent continuer à envoyer des requêtes pour un serveur même lorsque celui-ci ne fonctionne pas.

- **NFSv3** : Ajout de nouvelles fonctionnalités comme le support des fichiers de grande taille et une meilleure gestion des erreurs.

NFSv3 supporte les écritures asynchrones, ce qui permet au serveur de dicter les bonnes politiques de synchronisation des données. Les données seront synchronisées avant qu'une commande d'engagement de gestion des données soit établie. Cette conception permet une meilleure mise en mémoire tampon par rapport à la version NFSv2.

NFSv3 peut également traiter les erreurs plus efficacement et gérer des fichiers plus massifs. Il peut gérer des fichiers de 64 bits, ce qui signifie qu'un utilisateur peut accéder à environ 2 Go de contenu de fichier en moyenne.

- **NFSv4** : Amélioration de la sécurité avec Kerberos et meilleure performance grâce à une gestion optimisée des états des connexions.

NFSv4 est la version la plus récente du protocole NFS. Il peut fonctionner sur Internet et à travers les pare-feu. Il ne nécessite pas de service `rpcbind`, ce qui permet de l'exécuter dans davantage d'endroits.

Le protocole de contrôle de transmission (ou TCP) fonctionne avec cette version NFS. Le TCP fait le lien entre une application et une IP. Il garde un œil sur les segments de données et n'a besoin de recevoir les trames perdues dans l'ensemble TCP que lorsque quelque chose doit être envoyé une seconde fois.

Le serveur acceptera également les commandes du port TCP 2049. Ce port est l'un des ports les plus couramment utilisés que vous trouverez sur le marché. Il n'a pas à interagir avec le processus Daemon comme les options **`rpcbind`** et **`lockd`**.

Services nécessaire pour le fonctionnement NFS

- **nfs** : Le service nfs démarrera le serveur et les processus RPC nécessaires pour accepter les systèmes partagés.
- **nfslock** : Le service nfslock démarre les processus RPC et permet aux clients NFS de verrouiller les fichiers.
- **portmap** : Avec ce service, vous pouvez prendre les réservations de port des services locaux. portmap répondra aux messages indiquant que certains ports sont disponibles pour l'accès aux fichiers.
- **rpc.mountd** : processus qui reçoit les demandes de montage et confirme qu'un ordinateur peut atteindre les fichiers NFS auxquels on souhaite accéder.
- **rpc.nsfld** : Vous pouvez définir des versions et des protocoles NFS spécifiques que le serveur doit supporter ici.
- **rpc.lockd** : Avec cette commande, les fichiers sont faciles à verrouiller sur le serveur.
- **rpc.statd** : Le protocole Network Status Monitor démarre avec cette commande. Il notifie les clients lorsqu'un serveur redémarre, et s'assure que le serveur reste en ligne sans être éventuellement arrêté.

Composants Clés de NFS

- **Exportation des répertoires** : Les administrateurs configurent le fichier `/etc/exports` pour définir quels répertoires sont accessibles et par qui.
- **Montage des répertoires** : Les clients montent les répertoires distants avec la commande `mount`. Exemple : **`mount -t nfs serveur:/chemin/partage /chemin/local`**
- **Services Nécessaires** :
 - **nfsd** : Le démon qui gère les requêtes NFS.
 - **mountd** : Gère les demandes de montage des clients.
 - **rpcbind** : Mappe les services RPC vers les ports réseaux.

Avantages de NFS

- **Simplicité d'utilisation** : Intégration native dans les systèmes UNIX/Linux.
- **Flexibilité** : Permet un accès simultané aux fichiers partagés.
- **Interopérabilité** : Fonctionne avec différents OS (avec outils appropriés).

Limitations de NFS

- **Sécurité** : Les anciennes versions (NFSv2, NFSv3) manquent de mécanismes de sécurité robustes.
- **Performances** : Peut être moins efficace sur des réseaux lents ou encombrés.
- **Complexité de configuration** : Nécessite une bonne compréhension des fichiers de configuration et des droits d'accès.

Cas d'utilisation

- **Serveur centralisé** : Une entreprise utilise un serveur NFS pour stocker des documents accessibles à tous les employés.
- **Environnements académiques** : Les universités partagent des ressources communes comme les projets ou les données de recherche.
- **Infrastructure de développement** : Les équipes de développement accèdent à un espace de stockage centralisé.

Configuration de base

Côté Serveur :

- Installer le package NFS :
- - **sudo apt install nfs-kernel-server**
- Configurer le fichier /etc/exports :
- - **/chemin/partage client_ip(rw,sync,no_subtree_check)**
- Redémarrer le service :
- - **sudo systemctl restart nfs-kernel-server**

Côté Client :

- Installer le client NFS :
- - **sudo apt install nfs-common**
- Monter le répertoire :
- - **sudo mount serveur:/chemin/partage /chemin/local**

Bonnes pratiques

- **Sécuriser NFS** : Utiliser NFSv4 avec Kerberos pour un accès authentifié.
- **Sauvegardes** : Toujours sauvegarder les données partagées.
- **Réseaux fiables** : Utiliser NFS sur un réseau stable et rapide.

Sources

- <https://www.atera.com/fr/blog/what-is-nfs-understanding-the-network-file-system/>
- chatgpt