

Administration réseau

Accès aux fichiers distants

A. Guermouche

1. Introduction

2. NFS

3. SAMBA

Plan

1. Introduction

2. NFS

3. SAMBA

Accès aux fichiers distants

Différences avec le transfert de fichier :

- ★ l'accès aux fichiers distants est complètement transparent pour l'utilisateur
- ★ tout se passe comme si le système de fichier distant était local
- ★ l'utilisateur peut éditer le fichier, le modifier, ... ; les modifications seront répercutées sur le système fichier distant

Les deux principaux protocoles :

NFS. Network File System (Unix/Sun-RPC)

SMB. Sserver Message Block (issu du monde Microsoft)

Plan

1. Introduction

2. NFS

3. SAMBA

NFS : Network File System

Présenté par SUN en 1985 pour permettre à ces stations sans disque d'accéder à un système de gestion de fichiers distants (RFC 1904).

Utilise les appels de procédures distantes Sun-RPC (qui sont issues des travaux sur NFS)

- ★ à priori, les clients et serveur NFS devraient être des processus utilisateur s'exécutant au-dessus de RPC/XDR/UDP/IP.
- ★ en fait, le client et le serveur NFS s'exécutent dans le noyau
 - ▶ le client pour rendre transparent l'accès à un fichier via NFS
 - ▶ le serveur pour des raisons d'efficacité

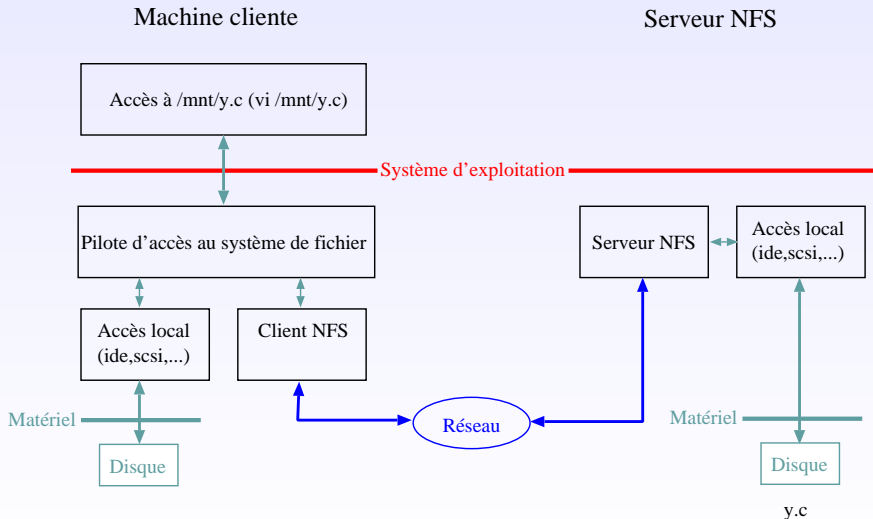
NFS : Network File System

Présenté par SUN en 1985 pour permettre à ces stations sans disque d'accéder à un système de gestion de fichiers distants (RFC 1904).

Utilise les appels de procédures distantes Sun-RPC (qui sont issues des travaux sur NFS)

- ★ à priori, les clients et serveur NFS devraient être des processus utilisateur s'exécutant au-dessus de RPC/XDR/UDP/IP.
- ★ en fait, le client et le serveur NFS s'exécutent dans le noyau
 - ▶ le client pour rendre transparent l'accès à un fichier via NFS
 - ▶ le serveur pour des raisons d'efficacité

Principe de fonctionnement



Les éléments d'accès aux fichiers

Processus utilisateur :
lecture/écriture dans un fichier

- ★ Manipule des descripteurs, chemins, déplacements

Système d'exploitation

Virtual File System (VFS)

EXT3

FAT

NFS

- ★ Manipule des Files, Dentries, Inodes, déplacements
- ★ Masque les différences à l'application (API uniforme, ...)

Matériel

Block device layer

IDE

SCSI

...

- ★ Manipule des blocs
- ★ Matériel-dépendant

Disque

Le choix entre NFS, EXT3, ... se fait lors de l'ouverture du fichier.

NFS et les RPCs

NFS repose sur les RPC (Remote Procedure Calls)

- ★ Utilisation du portmapper (programme portmap de Linux)
- ★ Portmapper = conversion des n° de prog RPC en numéro de port

Déroulement d'une RPC :

- ★ Serveur RPC :
 - Indique à portmap le port qu'il utilise et les numéros de programme RPC qu'il gère
- ★ Envoi d'une requête RPC par un client :
 - Il contacte portmap du serveur pour connaître le numéro de port du programme souhaité
 - Il envoie les données au port correspondant

Un serveur sans état

Dans un accès à un système de fichier local :

- ★ Les accès reposent sur un pointeur de fichier maintenu au niveau du système d'exploitation

NFS est basé sur une connexion réseau :

⇒ Probabilité d'une panne importante

Solution :

- ★ Le serveur NFS ne conserve aucune information sur les accès/opérations effectuées (fichiers ouverts, accès précédents au fichier, ...)
- ★ Le système d'exploitation du client NFS se charge de maintenir les informations concernant les fichiers

Intérêts :

- ★ Simplifie le redémarrage du serveur en cas de crash.

Sécurité

Principe d'authentification :

- ★ $(uid, gid)_{local}$ “mappé” sur $(uid, gid)_{distant}$
 - équivalence entre les droits locaux et les droits distants
- ★ Problème pour `root` :
 - ▶ Quels droits possède le `root` d'une machine cliente sur les fichiers exportés par un serveur NFS?
 - ▶ par défaut `root` (coté client) correspond à l'utilisateur `nobody` (coté serveur) pour des raisons de sécurité (sinon il faut mettre l'option `no_root_squash` dans `/etc/exports`)

Règle de non transitivité :

- ★ Si A exporte `/home` à B; Si B monte `A : /home` dans `/home2` et exporte `/home2` à C alors C n'aura pas accès au `/home` de A

Liens symboliques :

- ★ Les liens symboliques relatifs sont interprétés par rapport au système de fichier du client.

NFS en pratique (1/2)

Démons importants utilisés par NFS :

- portmap.** Gestion des connexions des applications utilisant le mécanisme de RPC.
- nfsd.** Authentification + Création, recherche, lecture et écriture de fichiers
- mountd.** Montage des systèmes exportés (mount et umount)
- statd.** Surveillance des nœuds du réseau (redémarrages. . .)
- lockd.** Section critique (lock les fichiers utilisés)

NFS en pratique (2/2)

coté client. le fichier `/etc/fstab` doit contenir le chemin vers le point de montage et le chemin sur le serveur NFS.

```
192.168.0.1:/home /nfs nfs defaults,noauto  
0 0
```

coté serveur. le fichier `/etc/exports` contient le chemin vers les dossiers à exporter ainsi

que la liste des machines autorisées à y accéder. `/home`
`192.168.1.0/255.255.255.0(rw,no_root_squash)`

Après chaque modification de `/etc/exports` il est nécessaire :

- ★ soit d'exécuter `exportfs` pour transmettre les modifications au serveur nfs
- ★ soit de relancer le serveur NFS

Plan

1. Introduction

2. NFS

3. SAMBA

SMB : Server Message Block

- ★ Protocole de Microsoft et Intel permettant le partage de ressources (disques, imprimantes, ...) à travers un réseau (1987)
- ★ SMB est prévu pour être utilisé au dessus de l'interface NetBIOS
 - ▶ Utilisation des noms NetBIOS (15 caractères + 1 pour le type)
 - ▶ Utilisation du mécanisme de datagramme de NetBIOS par *broadcast* comme service de nommage (nom → MAC, pas d'adresse de niveau 3)

Application		
SMB		
NetBIOS		
TCP/IP	NetBEUI	IPX/SPX
802.x	PPP	...

SMB (1/2)

- ★ Chaque machine client ou serveur possède un nom sur 15 caractères
- ★ SMB ajoute un 16ème caractère pour distinguer les serveurs de fichiers, les clients, les imprimantes, ...
- ★ Notion de domaine
 - ▶ un ensemble d'utilisateurs (avec nom et mot de passe) et de serveurs (avec des droits d'accès)
 - ▶ un *primary domain server* contient la base de données des utilisateurs et de leur mot de passe
- ★ Un serveur une ou plusieurs ressources
 - ▶ fichiers, imprimantes, ...
 - ▶ à chaque triplet (domaine, serveur, ressource) correspond un nom unique : `\\serveur\ressource`

SMB (2/2)

Deux niveaux de protection :

- ★ au niveau de chaque utilisateur : basé sur le nom des utilisateurs, permet de gérer l'accès aux ressources voire aux éléments d'une ressource
- ★ au niveau de chaque ressource : un mot de passe commun à tous les utilisateurs est associé à une ressource pour y autoriser l'accès

Résolution de noms : 4 méthodes utilisées

- broadcast.** résolution par diffusion d'une requête dans le réseau
- lmhost.** résolution en utilisant des associations prédéfinies entre noms NetBIOS et IP
- host.** utilisation de DNS
- wins.** utilisation d'un serveur WINS (*Windows Internet Name Server*). À chaque machine est associé un serveur WINS à qui elle envoie ses requêtes et auprès duquel elle s'enregistre.

SAMBA (1/2)

Samba : Implémentation de SMB sous UNIX qui permet le partage de ressources entre les mondes UNIX et Windows

Samba permet de :

- ★ partager un disque UNIX pour des machines Windows
- ★ accéder à un disque Windows depuis une machine UNIX
- ★ partager une imprimante UNIX pour des machines Windows
- ★ utiliser une imprimante Windows à partir d'un hôte Linux.

Le serveur Samba sur la machine Unix émule un domaine SMB

SAMBA (2/2)

Serveur Samba :

- ★ configuration via le fichier `/etc/smb.conf`
- ★ travail partagé par deux démons :
 - `smbd.` pour le service “serveur”
 - `nmbsd.` pour le service résolution des noms NetBIOS

Client :

- `smbpasswd.` permet de changer le mot de passe d'un utilisateur SMB
- `smbclient.` permet d'interroger un serveur Samba depuis UNIX
`smbclient //host/ressource` permet l'accès à la ressource

Possibilité de monter une partition Windows distante à l'aide de Samba → utiliser le système de fichier `smbfs`

Exemple : (extrait du fichier `/etc/fstab`)

```
//serveur/ressource /commun smbfs defaults 0 0
```