

# Activité : synchronisation distante de fichiers

## Présentation

Les solutions de mise en oeuvre de **redondance de services** nécessitent la **duplication de données** sur des serveurs distincts éventuellement éloignés sur des sites distants.

Certaines solutions de sauvegarde en réseau utilisent par ailleurs des outils de duplication de données pour réaliser des **sauvegardes incrémentales** d'informations ou pour diffuser sur plusieurs machines le contenu d'un répertoire de référence.

Vous allez mettre en oeuvre le logiciel de synchronisation de fichiers en ligne de commande **Rsync**. Celui-ci :

- fonctionne de manière **unidirectionnelle** c'est à dire qu'il copie les fichiers d'un dossier vers un autre,
- **synchronise** le répertoire de destination en fonction du dossier source en faisant une comparaison simple des fichiers entre source et destination,
- ne prend en compte que les **modifications** de la source de données.

Rsync permet alors :

- de réaliser des **sauvegardes incrémentales** de données,
- ou pour **diffuser** sur plusieurs machines le contenu d'un répertoire de référence.

Vous allez mettre en place une **synchronisation de données** :

- entre deux **deux dossiers** situés sur **deux serveurs** Linux Debian différents,
- entre un **dossier** situé sur un **serveur Linux** et un **dossier** situé sur **serveur Windows**.

Pour des raisons de **sécurité**, l'accès à distance en SSH avec le compte **root** est **interdit**. Pour le permettre, le fichier **/etc/ssh/sshd\_config** doit être modifié :

Sous la ligne suivante :

```
# PermitRootLogin prohibit-password
```

Ajoutez la ligne par (sans le #):

```
PermitRootLogin yes
```

Puis redémarrez le service SSH :

```
# service ssh restart
```

## Première maquette à réaliser

- utiliser **deux serveurs** Linux Debian.
- **synchroniser** de manière **unidirectionnelle**, dans le sens Serveur1 vers Serveur2, d'un dossier situé dans le dossier du compte **root** de chaque serveur,
- créer un script permettant de faire cette synchronisation toutes les 5 minutes et utilisant une **authentification par clé SSH**.  
===== Installer Rsync =====

```
# apt update
# apt install rsync
```

## Découvrir Rsync

**Syntaxe de base :**

```
rsync [option] /chemin/vers/source /chemin/vers/destination
```

**Copie distante en utilisant SSH :**

```
rsync [option] -rsh=ssh /chemin/vers/source [user]@[host]:/chemin/vers/destination
```

Pour transférer des données vers un serveur distant, le serveur **OpenSSH** doit être **installé** :

```
# apt install openssh-server
```

**Tableau des options**

OPTIONS	ACTION
-r	visite récursive des répertoires
-l	copie les liens symboliques comme liens symboliques
-p	préserve les permissions
-t	préserve les dates
-g	préserve le groupe
-o	tous les fichiers des *-from/filtres sont séparés par 0
-D	préserve les périphériques (root uniquement)
-a	mode archivage; identique à -rlptgoD (pas -H)
-v	De nombreuses informations sont affichées lors du transfert.
-z	Compresse le fichier pour un meilleur transfert par le reseau.
-delete	ATTENTION ! Si le fichier (ou son contenu) n'existe plus dans "source", il sera supprimé dans "destination".

**Exemples**

- Copier intégralement le fichier ou le répertoire source et son contenu vers destination avec compression des données

```
rsync -avz /root/dossier /root/sauvdossier
```

- Copier intégralement le fichier ou le répertoire source et son contenu vers destination sur un serveur distant avec compression des données <code> rsync -avz -rsh=ssh /root/dossier [root]@[@ip]:/root/sauvdossier </code>

Il sera nécessaire d'indiquer le mot de passe du compte

Par défaut le compte root ne peut se connecter à distance au szerveur en SSH. POur changer cela, il faut modifier le fichier /etc/sshd/sshd\_config

**Authentification par clé SSH**

Afin de ne pas avoir à fournir le **mot de passe** à chaque **exécution** de rsync, le serveur source va s'authentifier sur le serveur destination avec une clé SSH.

Il faut définir le serveur **source** comme un **client SSH de confiance**. Pour cela :

- créer un couple de clefs publique et privée sans mot de passe,
- ajouter cette toute récente clef publique sur le serveur de destination.

```
# ssh-keygen -t rsa
# ssh-copy-id -i /root/.ssh/id_rsa.pub btssio@AdresseIPserveurDestination
```

Vérifiez qu vous pouvez vous connecter sans mot de passe :

```
#ssh btssio@AdresseIPserveurDestination
```

Tutoriel : [Configurer une authentification avec un couple de clés privée/publique SSH](#)

## Script de sauvegarde

Pour automatiser la sauvegarde, vous allez créer un script de sauvegarde simple `/root/script/rsync.sh` contenant :

```
rsync -avz -rsh=ssh /root/dossier [btssio]@[@ip]:/home/btssio/sauvdossier
```

## Sauvegardes automatisées depuis le cron du serveur

Il est **conseillé** d'effectuer les sauvegardes à des moments de la journée où les serveurs sont **peu sollicités**. Pour cela, il faut inscrire des appels à la commande de sauvegarde dans le **cron** de l'utilisateur **root** du serveur.

```
# crontab -e
15 * * * * /root/scripts/rsync.sh
```

## Deuxième maquette à réaliser

- utiliser **un de vos serveurs Linux Debian** et le serveur **DC1-0870019Y**.
- **synchroniser** de manière **unidirectionnelle**, dans le sens Serveur Debian vers le serveur DC1-0870019Y, d'un dossier situé dans le dossier du compte root du serveur Debian dans votre dossier personnel du serveur Windows,
- créer un script permettant de faire cette synchronisation toutes les 5 minutes et utilisant une **authentification par clé SSH**.

## Configuration du serveur Linux

Les serveurs Microsoft Windows utilisent le **protocole SMB (Server Message Block)** pour **partager** des fichiers et des imprimantes. Ce protocole a été créé à l'origine par IBM, puis repris par Microsoft. Ce protocole a ensuite été renommé en **protocole CIFS (Common Internet FileSystem)**. Lors de la sortie de Windows Vista, Microsoft a repris la dénomination SMB pour sa nouvelle version SMB 2. **SMB et CIFS** désignent donc, pour l'essentiel, le même protocole.

Pour accéder à ces serveurs Windows, il faut utiliser un **client SMB**. Le projet Samba consiste à fournir les paquets logiciels pour les distributions Linux afin de :

- pouvoir créer un **serveur SMB/CIFS**,
- disposer d'un **client SMB/CIFS**.

## Installation d'un client SMB/CIFS

```
# apt install smbclient
```

### lister les partages d'un serveur Windows

```
# smbclient -L nomserveur -U login_Windows
```

Il vous demandera de taper le **mot de passe** pour le domaine Windows

## Accéder à un partage et aux fichiers

Exemple pour télécharger un fichier situé dans le dossier de la classe :

- serveur : DC1-serveur
- partage : B2classe

```
# smbclient //DC1-serveur/B2classe -U login_Windows
smb: \> ls
smb: \> cd répertoire_distant
smb: \> get fichier.txt
smb: \> quit
```

Le fichier téléchargé est ensuite **disponible** dans le dossier root.  
La commande **put** permet de déposer un fichier **présent** dans le dossier root sur le **serveur**;

### Monter un partage ponctuel sur le client Debian

Il est possible de monter sur un client Linux Debian, un partage situé sur un serveur Windows en installant le paquet smbfs :

```
# apt install cifs-utils
```

- **Création** du point de montage du dossier **B2classe** dans l'arborescence du client Debian en **modification** :

```
# mkdir /mnt/B2classe
```

- montage du dossier partagé B2classe en mise à jour

```
# mount -t cifs -o user=login_Windows,password=password,rw //DC1-serveur/B2classe /mnt/B2classe
```

Vous pouvez alors, depuis votre client Debian, parcourir le partage situé sur le serveur Windows, en lecture et en écriture.

Pour démonter le montage :

```
# umount /mnt/B2classe
```

### Monter un partage permanent sur le client Debian

Pour rendre ce montage permanent; il faut mettre à jour le fichier **/etc/fstab** de montage des partitions.

**Attention** : la modification incorrecte du fichier **/etc/fstab** rendra vos partitions inutilisables !

- copie de sauvegarde du fichier **/etc/fstab** `# sudo cp /etc/fstab /etc/fstab_sauvegarde`
- ajout en fin de fichier d'une ligne de ce type `DC1-serveur/B2classe /mnt/B2classe/ cifs  
username=login_Windows,password=password,uid=root,gid=root,fmask=777,dmask=777 0 0`

Explications :

- //DC1-serveur/B2classe : nom du dossier partagé auquel vous voulez accéder.
- /mnt/B2classe/ : point de montage sur le client Debian (Le dossier doit exister).
- username : identifiant utilisateur Windows.
- password : mot de passe de l'identifiant utilisateur Windows.
- uid et gid : nom d'utilisateur Linux et nom du groupe Linux.
- droits fmask (fichiers) et dmask (dossiers) à 777 pour permettre d'accéder à tous les fichiers du disque réseau.

\* Redémarrez votre client Debian pour vérifier que le montage est bien automatiquement créé.

**A faire** : modifiez votre script pour avoir une sauvegarde de votre dossier Situé sur le client Debian, sur le serveur Windows.

===== Ressources ===== \* <http://debian-facile.org/doc:reseau:rsync> \* <https://wiki.debian.org/fr/RsyncSnapshots> \* <http://www.mathrice.org/rencontres/mars.2007/presentations/rsync-unison.pdf>

From:

/ - **Les cours du BTS SIO**

Permanent link:

</doku.php/sisr3/arsync>

Last update: **2017/10/18 15:29**

