|  |  |
| --- | --- |
|  | **Service de sauvegarde** |

## Table des matières

# rsync

[rsync 1](#_Toc85370859)

[Présentation 2](#_Toc85370860)

[Installation 3](#_Toc85370861)

[Utilisation 3](#_Toc85370862)

[Deux exemples de base 4](#_Toc85370863)

[Subtilité des chemins 5](#_Toc85370864)

[Utilisation de *rsync* en local 5](#_Toc85370865)

[Utilisation de rsync en réseau 6](#_Toc85370866)

[Utilisation des clés publique/privée pour la sauvegarde par ssh 6](#_Toc85370867)

[Utilisation d'un port ssh différent de celui d'origine 6](#_Toc85370868)

[Annexe 01 Exemple pratique 7](#_Toc85370869)

[Annexe 02 Quelques options intéressantes 8](#_Toc85370870)

[Affichage de la progression lors du transfert de données 8](#_Toc85370871)

[Utilisation des options --inclure et --exclure 8](#_Toc85370872)

[Utilisation de l'option --delete 8](#_Toc85370873)

[Définition de la taille maximale des fichiers à transférer 8](#_Toc85370874)

[Suppression automatique des fichiers source après un transfert réussi 9](#_Toc85370875)

[Essai (test de la sauvegarde) 9](#_Toc85370876)

[Définition de la limite de bande passante et transférer le fichier 9](#_Toc85370877)

## Présentation

*rsync* (pour *remote synchronization* ou synchronisation à distance), est un logiciel de synchronisation de fichiers. Il est fréquemment utilisé pour **mettre en place des systèmes de sauvegarde distante**.

*rsync* est une application qui permet de faire des **sauvegardes locales ou à distance d'une arborescence de fichiers et répertoires**. Pour ce faire, *rsync* peut utiliser un **tunnel SSH** afin d'effectuer des sauvegardes au moyen d’une connexion réseau d'un hôte à une autre de manière à sécuriser les données transmises.

rsync fonctionne par une **copie partielle des données**.   
Il ne recopie que ce qui a été **modifié depuis la dernière sauvegarde** (nouveaux fichiers/répertoires ou fichiers/répertoires modifiés).   
Il s’agit donc d’une **sauvegarde incrémentielle**.   
Le début du transfert commence donc par l'**envoi** **de la liste des fichiers ainsi que plusieurs informations** relatives à ceux-ci, puis les données elles même.

*rsync* travaille donc de **manière unidirectionnelle** c'est-à-dire qu'il **synchronise**, **copie** ou **actualise** les données d'une source (locale ou distante) vers une destination (locale ou distante) en ne transférant que les octets des fichiers qui ont été modifiés.   
Cela signifie qu'en une commande, la **synchronisation ne peut se faire que dans un sens**.

Quelques avantages et fonctionnalités de l’utilitaire rsync ...

* rsync **copie et synchronise efficacement** les fichiers vers ou depuis un système distant;
* rsync **prend en charge la copie des liens**, **des périphériques** (devices), **des propriétaires**, **des groupes** et **des** **autorisations** ;
* rsync est **plus rapide que scp** (secure copy) car rsync utilise un protocole de mise à jour à distance qui permet de **transférer uniquement les différences entre deux ensembles de fichiers**.   
  La **première fois**, il **copie l'intégralité** du contenu d'un fichier ou d'un répertoire de la source à la destination, mais **les fois ultérieures**, rsync **copie uniquement les blocs et octets modifiés** vers la destination.
* rsync **consomme moins de bande passante** car il utilise la méthode de **compression et de décompression** lors de l'envoi et de la réception de données aux deux extrémités.

D'un point de vue de l'entreprise, un **serveur de synchronisation est utile pour centraliser les sauvegardes** sur un même hôte équipé accessoirement d'un enregistreur de bande.

## Installation

Comme d'habitude, on commence par la mise à jour des paquets …  
**>>** **sudo** **apt update &&** **sudo** **apt upgrade**

pour ensuite installer rsync…  
>> **sudo apt install rsync**

Pour vérifier la version rsync installée …  
>> **sudo** r**sync --version**  
**rsync version 3.1.3 protocol version 31**

Afin de vérifier si le service est démarré …  
>> **sudo** **systemctl status rsync.service**   
**● rsync.service - fast remote file copy program daemon**  
 **Loaded: loaded (/lib/systemd/system/rsync.service; enabled; vendor preset: en**  
 **Active: inactive (dead)**  
 **Docs: man:rsync(1)**  
 **man:rsyncd.conf(5)**  
**...**

Une fois installé, on peut utiliser rsync sur n'importe quel hôte à lequel on peut accéder à l’aide d’une invite de commande (*shell*) distante.

Pour les transferts à distance, rsync utilise ssh pour ses communications, mais il peut être configuré pour utiliser une invite de commande distante différente par défaut, tel que rsh ou remsh.

## Utilisation

La **commande de base** pourrait se lire comme suit…

>> **rsync <Chemin source> <Chemin cible>**

* **<Chemin source>** est le chemin du répertoire qui servira de source, de référence ;
* **<Chemin cible>** est le chemin du répertoire qui servira de cible.

Le protocole de mise à jour à distance rsync lui **permet de ne transférer que la différence entre deux jeux de fichiers** à travers le réseau, en utilisant un algorithme de recherche de somme de contrôle.

Ce protocole établit ses connexions sur le **port TCP 873 par défaut**.

La synchronisation fonctionne selon deux modalités ...

* en **mode local**   
  (sur le même système de fichiers ou sur un répertoire réseau monté en local), où le programme client se charge de comparer les deux versions des répertoires et effectuer la synchronisation ;
* en **mode distant**   
  le programme client contacte le programme serveur, habituellement grâce au protocole SSH, ce qui permet une communication sécurisée à travers Internet.

## Deux exemples de base

Considérant la commande suivante …  
**>>** **rsync -Haurovz <Chemin source> <Chemin cible>**

Description des **options Haurovz** …

* **-H** ou **--hard-links**  
  **préserve les liens physiques** (*hard links*) ;
* **-a** ou **--archive**  
  indique le **mode archivage**   
  Ce mode impose la récursivité permettant ainsi de **garder les propriétés des fichiers intactes**   
  (autorisations, propriété, lien symbolique) et de ne pas suivre les liens symboliques ;
* **-u** ou **--update**  
  indique la **mise à jour des fichiers modifiés** et **non la copie de tous les fichiers** ;
* **-r** ou **--recursive**  
  impose le **mode récursif**, soit la sauvegarde du contenu du **répertoires et ses sous-répertoires** ;
* **-o** ou **--owner**  
  permet de de **ne pas modifier le propriétaire des fichiers**   
  (cela ne fonctionne que si on effectue la **sauvegarde en tant qu’usager root**) ;
* **-z** ou **--compress**  
  **compresse les données** lors du transfert.

Considérant la commande suivante …  
**>>** **rsync -vrazh <Chemin source> <Chemin cible>**

À quoi correspondent les **options vrazh** …

* **-v** ou **--verbose**   
  **augmente la verbosité** ;
* **-r** ou **--recursive**  
  impose le **mode récursif**, soit la sauvegarde du contenu du **répertoires et ses sous-répertoires** ;
* **a** ou **--archive**  
  indique le **mode archivage** ;  
  Le mode archive permet de copier des fichiers de **manière récursive** et il préserve également les liens **symboliques**, **les autorisations** de fichier, la **propriété des** utilisateurs et des groupes et **les horodatages**
* **--z** ou **--compress**  
  **compresse les données** lors du transfert ;
* **-h** ou **--human-readable**  
  affiche les numéros dans un **format lisible par l'homme**.

## Subtilité des chemins

Il convient d'être vigilant dans l'utilisation ou non de la barre oblique ( / ) dans le chemin de la source.   
Ainsi, les deux commandes suivantes ne sont pas équivalentes …  
**>> rsync <Chemin source> <Chemin cible>/**  
**>>** **rsync <Chemin source>/ <Chemin cible>/**

L'absence de la barre oblique ( / ) a son importance …

* **sans** / *rsync* **recopie le** **répertoire** **et son contenu**;
* **avec** / *rsync* ne **recopie** **que le contenu du répertoire**.

En effet, la première commande va créer le répertoire source dans le dossier destination en ajoutant donc un niveau dans l'arborescence.  
La seconde commande copie le contenu du répertoire source dans le répertoire cible.

Enfin, il faut noter que l'utilisation ou non d'une barre oblique (/ slash) final dans le chemin de la destination n'a aucune incidence.   
Les deux commandes suivantes sont donc équivalentes …  
**>> rsync <Chemin source> <Chemin cible>**  
**>> rsync <Chemin source> <Chemin cible>/**

## Utilisation de *rsync* en local

Voici un premier exemple de sauvegarde d’un **dossier d'hébergement dans un répertoire de sauvegarde** …  
**>>** **rsync -Haurov /var/www/profsavard.info/ /home/archives/**

Cette commande va donc sauvegarder le répertoire */var/www/cmaisonneuve.qc.ca/* dans le répertoire */home/archives/*.

* **-v** (ou --verbose) pour *verbose*,  
  permet d’**afficher** **les informations lors de la copie**

Si on veut **supprimer de la sauvegarde les fichiers qui ont été supprimés du répertoire source**.  
Il faut ajouter l'option --delete …  
**>>** **rsync -Haurov --delete /var/www/profsavard.info/ /home/archives/**

Si un **fichier est malencontreusement supprimé du serveur**, il le sera donc à la prochaine sauvegarde, il est donc possible de **stocker les fichiers supprimés dans un autre répertoire par sécurité** ...  
**>>** **rsync -Haurov --delete --backup --backup-dir=/home/archives/deleted/   
 /var/www/profsavard.info/ /home/archives/**

Il est aussi possible d'**exclure** **un répertoire de la sauvegarde**, exemple un répertoire où se trouveraient des fichiers de cache ...  
**>>** **rsync -Haurov --exclude=/var/www/profsavard.info/cache/ /var/www/profsavard.info/ /home/archives/**

Il existe aussi l'option --*progress* qui **affiche la progression du transfert** ...  
**>>** **rsync -Haurov --progress /var/www/profsavard.info/ /home/archives/**

## Utilisation de rsync en réseau

Le point intéressant avec rsync, est de pouvoir sauvegarder sur une hôte distant, pour cela il faut spécifier notre serveur cible (pour cet exemple 123.123.123.123) avec un utilisateur spécifique (pour cet exemple : root) dans le répertoire /home/archives …  
**>>** **rsync -Haurov /var/www/profsavard.info/ root@123.123.123.123:/home/archives/**

Cette **sauvegarde utilise ssh**, rsync demandera donc le **mot de passe de l'utilisateur distant** pour effectuer sa sauvegarde.

### Utilisation des clés publique/privée pour la sauvegarde par ssh

Il faut, dans un premier temps, **générer la paire de clés privée et publique**.   
Une fois les clés installées, il suffit d’utiliser cette commande pour la sauvegarde …  
**>>** **rsync -Haurov -e "ssh -i /root/.ssh/id\_dsa" /var/www/profsavard.info/ root@123.123.123.123:/home/archives/**

Cela **permettra de lancer votre commande en tâche planifiée crontab** par exemple.

### Utilisation d'un port ssh différent de celui d'origine

Si le serveur recevant les sauvegardes dispose d'un port ssh n'était pas celui d'origine (22) mais par exemple 2222, utilisez la commande suivante …  
**>>** **rsync -Haurov -e "ssh -p 2222" /var/www/maisonneuve.info/ root@123.123.123.123:/home/archives/**

# Annexe 01 Exemple pratique

Dans un premier temps, on crée répertoires et fichiers pour la synchronisation …

**>>** **cd ~**   
**>>** **mkdir** **dir{1,2}**  
**>>** **touch dir1/fichier{1..100}**   
Il existe maintenant un répertoire nommé dir1 contenant 100 fichiers vides.

Il est important de toujours tester les arguments avant d'exécuter une commande rsync.   
rsync fournit une méthode pour ce faire en passant les options -n (ou --dry-run).   
L’option -v (--verbose) est également nécessaire pour obtenir la sortie appropriée …  
**>>** **rsync -anv dir1/ dir2**

# Annexe 02 Quelques options intéressantes

## Affichage de la progression lors du transfert de données

Pour afficher la progression lors du transfert des données d'un hôte vers un autre hôtee, il est possible d’utiliser l'**option --progress**.   
rsync affiche les fichiers et le temps restant pour terminer le transfert …  
**>>** **rsync -avzhe ssh --progress /var/www/maisonneuve.info/** [**root@123.123.123.123:/home/archives/**](mailto:root@123.123.123.123:/home/archives/)

## Utilisation des options --include et --exclude

Ces deux options permettent d'inclure et d'exclure des fichiers lors de la synchronisation.   
Ces options précisent les fichiers ou répertoires que l’on souhaite inclure lors de la synchronisation et d'exclure les fichiers et dossiers que l’on ne souhaite pas transférer  
Pour cet exemple, la commande rsync inclura uniquement les fichiers et le répertoire qui commencent par la lettre R et exclura tous les autres fichiers et répertoires.

**>>** **rsync -avze ssh --include 'R\*' --exclude '\*' /var/www/maisonneuve.info/** [**root@123.123.123.123:/home/archives/**](mailto:root@123.123.123.123:/home/archives/)

## Utilisation de l'option --delete

Si un fichier ou un répertoire n'existe pas à la source, mais existe déjà à la destination, on souhaite peut-être supprimer ce fichier ou répertoire existant sur la cible pendant la synchronisation.  
L’utilisation de l’**option --delete** afin de supprimer les fichiers qui ne se trouvent pas dans le répertoire source.

La source et la cible sont synchronisées.   
On crée maintenant un nouveau fichier test.texte sur la cible.

**>>** **touch test.texte**  
**>>** **rsync -avz --delete** [**root@123.123.123.123:/home/archives/**](mailto:root@123.123.123.123:/home/archives/)**.**

## Définition de la taille maximale des fichiers à transférer

Il est possible de spécifier la taille de fichier maximale à transférer ou à synchroniser.   
On peut le faire avec l'**option --max-size**.   
Pour cet exemple, la taille de fichier maximale est de 200 Ko, donc cette commande ne transférera que les fichiers égaux ou inférieurs à 200 Ko …  
**>>** **rsync -avzhe ssh --max-size='200k' //var/www/maisonneuve.info/ root@123.123.123.123:/home/archives/**

## Suppression automatique des fichiers source après un transfert réussi

L’exemple suivant implique un serveur Web principal et un serveur de sauvegarde de données, qu’il existe une sauvegarde quotidienne et qu’une sauvegarde sur le serveur de sauvegarde a déjà été effectuée.  
Pour la suite des choses, on ne désire pas conserver cette copie locale de la sauvegarde sur le serveur Web.

On ne souhaite pas supprimer ces fichiers de sauvegarde locaux manuellement suite à un transfert réussi,   
La suppression automatique peut être effectuée en utilisant l’**option --remove-source-files** …  
**>>** **rsync --remove-source-files -zvh /var/www/maisonneuve.info/** [**root@123.123.123.123:/home/archives/**](mailto:root@123.123.123.123:/home/archives/)

## Essai (test de la sauvegarde)

Afin d’effectuer un test avant d’exécuter une sauvegarde, l'utilisation l’**option** **--dry-run** n'apportera aucune modification. La commande va afficher la sortie de la commande et en observant si la sortie affiche exactement la même chose que l’on souhaite faire, on pourra supprimer l'option --dry-run de la commande et l’exécuter sur le terminal …  
**>>** **rsync --dry-run --remove-source-files -zvh** **/var/www/maisonneuve.info/ root@123.123.123.123:/home/archives/**

## Définition de la limite de bande passante et transférer le fichier

Si on souhaite définir la limite de bande passante lors du transfert de données d'un hôte vers un autre hôte à l'aide de l'**option** **--bwlimit**. Cette option nous aide à limiter la bande passante d'E/S …  
**>> rsync --bwlimit=100 -avzhe ssh** **/var/www/maisonneuve.info/ root@123.123.123.123:/home/archives/**