Sauvegarde de données sous Linux

La gestion de la sauvegarde à bien évoluer sur les systèmes Linux.
Des outils qui géraient dans le meilleur des cas un lecteur de bande local, jusqu'aux outils modernes sophistiqués et aux logiciels de sauvegarde commerciaux, l'éventail est large.
L'essentiel est de connaître les moyens disponibles et d'adapter sa stratégie de sauvegarde à ses besoins en fonction du temps et de l'argent qu'on est prêt à investir dans la sauvegarde.

Je vais donc présenter quelques moyens pour effectuer des sauvegardes.

1. Les utilitaires d'archivage

Les utilitaires d'archivage permettent de réaliser les sauvegardes les plus simples et grâces à cette simplicité, sans doute les plus fiables.
Leur principe est simple elles envoient un ensemble de fichiers (en générale une arborescence de répertoires) vers un fichier, qu'il s'agisse d'un fichier ordinaire ou d'un fichier spécial qui désigne un périphérique de stockage.

1.2. la commande tar

La commande `tar`, d'usage universel dans les environnements Linux est à connaître absolument.
Sa richesse fonctionnelle peut impressionner, mais si la commande `tar` présente de très nombreuses options.
syntaxe de la commande `tar` pour créer une archive

```
tar action compression verbosité -f fichier_archive répertoire
```

syntaxe de la commande `tar` pour lister ou extraire une archive

```
tar action compression verbosité -f fichier_archive
```

Commande tar - quelques options et paramètres :
action :

- -c : Créer une archive.
- -t : Liste le contenu d'une archive existante.
- -x : Extrait le contenu d'une archive existante dans le répertoire courant.

compression :

- pas d'option : Pas de compression sur l'archive manipulée.
- -z : Compression au format gzip de l'archive manipulée.
- -j : Compression au format bzip2 de l'archive manipulée.

verbosité :

- pas d'option : Pas de verbosité, affichage minimum.
- -v : verbosité, affichage détaillé.

fichier_archive : Le fichier qui reçoit ou héberge l'archive.
répertoire : Dans le cadre d'une création d'archive, désigne le répertoire à partir duquel l'archive est créée.

Même si ce n'est pas obligatoire, il d'usage de mettre une extension “.tar” au fichiers contenant une archive tar, suivi d'une extension liée au mode de compression “.gz” ou “.bz”.

Exemple d'utilisation :

```
~# echo test> test
~# echo test> toto
~# ls
test toto
~# tar -cvzf
~# tar -cvzf test.tar.gz test toto
test
toto
~# ls
test  test.tar.gz  toto
```

Si la commande `tar` est employée pour créer une archive sur bande magnétique et non sur disque.
Il est recommandé de ne pas utiliser d'option de compression.
Le format compressé empêcherait une récupération partielle des données en cas de détérioration de la bande.

1.2. La Commande cpio

Cette commande permet de réaliser des archives non compressées d'un ensembles de fichiers et répertoires.
La commande `cpio` est de moins en moins utilisé, car en effet cette commande n'est pas très intuitif, le problème avec cette commande est qu'on ne peut pas sélectionné les éléments à sauvegarder en tant que paramètre comme le fait la commande `tar`.
Il faut lui indiqué les éléments sous forme de liste de fichiers sur son entrée standard.
La liste de fichiers à sauvegarder est presque toujours fournie par redirection de la commande `find`.

Donc, cette commande sert dans le cas ou l'on veut faire des sauvegardes très sélectives.

Syntaxe de la commande `cpio` pour créer une archive

```
find répertoire critère -print | cpio options > fichier_archive
```

Syntaxe de la commande `cpio` pour lister ou extraire une archive

```
cpio options < fichier_archive
```

Commande cpio - quelques options et paramètres :
répertoire : le répertoire de base à partir duquel se fait la recherche.
critère : critères de recherche selon la syntaxe de la commande `find`.
options :

- -o : mode copy-out.
  Indique qu'on est en mode de création d'archive.
  Exclusif des options i et f)
- -t : Associée à l'option i, liste le contenu d'une archive existante.
  Exclusif de l'option o.
- -i : mode copy-in. Indique qu'on est en mode d'extraction ou de consultation d'archive.
  Exclusif de l'option o.
- -v : facultatif : rend la commande verbeuse.

fichier_archive : le fichier qui recevra l'archive.

Exemple d'utilisation :

```
~# echo test
test
~# echo test> test
~# echo test> toto
~# ls
test  toto
~# find t* -print | cpio -o >archive.cpio
1 bloc
~# ls
archive.cpio  test  toto
```

2. Les sauvegardes au niveau filesystem

2.1. filesystem ext

Les utilitaires `dump` et `restore`permettent de réaliser des sauvegarde incrémentales et la restauration d'un filesystem complet.

Exemple d'utilisation de la commande `dump` :

```
dump 0 -f /backup/sda1-0.dump /dev/sda1 2>>/var/log/backup-sda.log
```

Dans cette exemple, le niveau de sauvegarde est 0, il correspond à une sauvegarde complète, alors que les autres niveaux *n* correspondent à la sauvegarde des fichiers qui ont été modifiés depuis la *nieme* sauvegarde.
-f indique le fichier de destination de la sauvegarde.

Exemple d'utilisation de la commande `restore` :

```
~# mount /dev/sda1 /mnt/data
~# cd /mnt/data
~# restore -rvf /backup/sda1-0.dump
```

L'option r indique qu'on réalise bien une restauration et pas une autre opération comme une comparaison, v rend la commande verbeuse et f annonce le fichier de sauvegarde à restaurer.
La restauration des fichiers se fait dans le répertoire courant.

2.3. filesystem xfs

Les utilitaires `xfsdump` et `xfsrestore` permettent de réaliser des sauvegardes incrémentales et la restaurations d'un filesystem xfs complet.

Exemple d'utilisation de la commande `xfsdump` :

```
xfsdump sda1-0.dump /dev/sda1
```

L'option `f` annonce le fichier de destination de la sauvegarde, `sda1-0.dump` est le fichier cible et `sda1` le filesystem à sauvegarder.

Exemple d'utilisation de la commande `xfsrestore` :

```
xfsrestore -f sda1-0.dump data
```

L'option `f` annonce le fichier contenant la sauvegarde et `data` est le répertoire de destination de la restauration.

3. Les logiciels de sauvegarde

AMANDA - (Advanced Maryland Automatic Network Disk Archiver) est une solution libre sous licence GNU GPL de sauvegardes informatiques.
Il est utilisé pour sauvegarder via le réseau un ensemble de postes clients et de serveurs (Unix, Linux, et Windows via Samba ou Cygwin).
Source : Wikipédia.org

Bacula - est un logiciel de sauvegarde, de vérification et de restauration de fichiers en mode client-serveur dans un réseau, sous licence GNU GPL.
Ses composants serveurs tournent sur Linux, Solaris ou FreeBSD.
Ses composants clients existent pour de nombreuses autres plateformes, dont MacOS et Windows.
Source : Wikipédia.org

BackupPC - est un logiciel libre de sauvegarde de système de fichier publié sous licence GPL.
Il est utilisé pour sauvegarder sur disque un ensemble de postes clients et de serveurs, sous Unix, Linux, Windows ou Mac OS X.
Les protocoles utilisables pour les transferts sont : SMB, tar over SSH/rsh/nfs et rsync.
Il ne nécessite l'installation d'aucun logiciel client sur les machines à sauvegarder.
Il possède une interface web pour lancer des sauvegardes ou restaurer des fichiers.
Il est également possible de sauvegarder des bases de données via un script shell lancé avant la

sauvegarde.
Source : Wikipédia.org

Les logiciels commerciaux - La plupart des grands éditeurs de logiciels de sauvegarde supportent souvent en option la sauvegarde système Linux.
Il faudra alors installer sur chaque système un agent de sauvegarde qui permettra de renvoyer les données vers le serveur de sauvegarde.

From:
http://www.ksh-linux.info/ - **Know Sharing**

Permanent link:
http://www.ksh-linux.info/systeme/sauvegarde-de-donnees-sous-linux

Last update: **12/11/2016 20:07**