

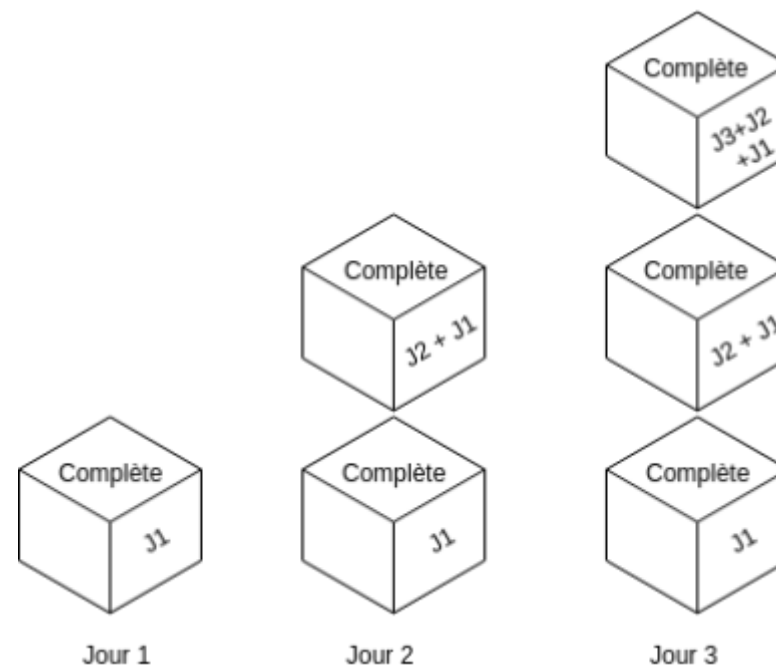
Dernière mise-à-jour : 2020/01/30 03:28

LSF111 - Gestion de l'Archivage et de la Compression

Types de Sauvegardes

Sauvegarde complète

Dans ce cas on sauvegarde tous les fichiers et dossiers, toute la partition ou tout le disque :



La restauration nécessite :

- La dernière sauvegarde complète.

Avantage :

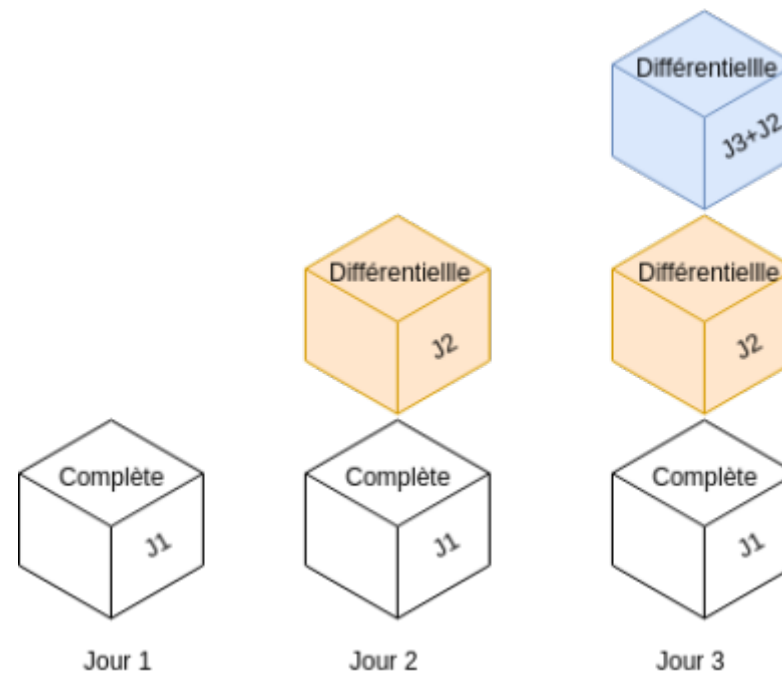
- Restauration plus rapide et plus simple que les restaurations à partir de sauvegardes différentielles et incrémentales.

Inconvénient :

- Nécessite plus d'espace de stockage que les sauvegardes différentielles et incrémentales.

Sauvegarde différentielle

Dans ce cas on sauvegarde des données modifiées ou nouvellement ajoutées sur la base de la **dernière sauvegarde complète** :



La restauration nécessite :

- La dernière sauvegarde complète,
- La dernière sauvegarde différentielle.

Avantage :

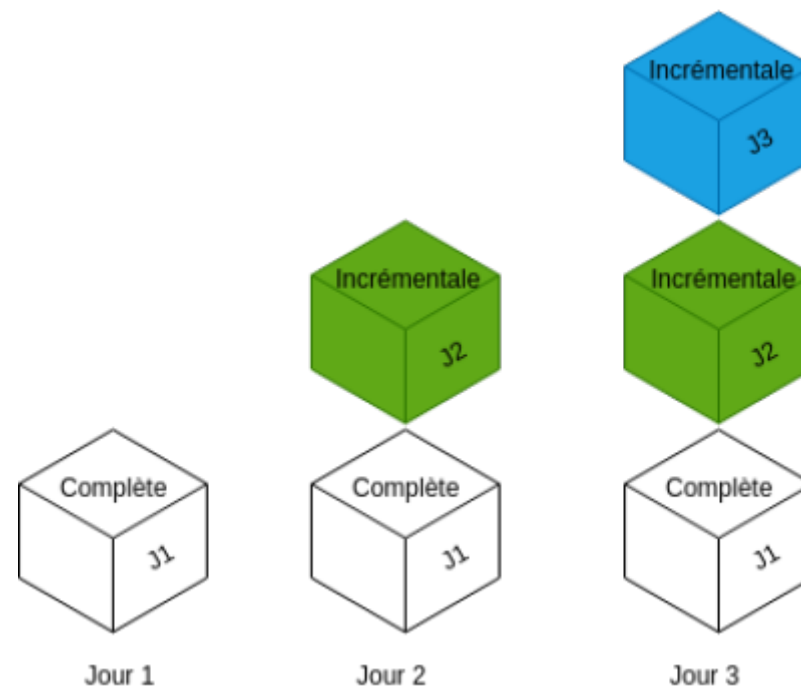
- Restauration plus rapide que la restauration à partir d'une sauvegarde incrémentale.

Inconvénient :

- Nécessite plus d'espace de stockage que la sauvegarde incrémentale.

Sauvegarde incrémentale

Dans ce cas, on ne sauvegarde que ce qui a changé depuis la dernière sauvegarde quelle qu'elle soit (complète, différentielle ou incrémentale) :



La restauration nécessite :

- La dernière sauvegarde complète,
- Toutes les sauvegardes incrémentales effectuées depuis, et dans l'ordre.

Avantage :

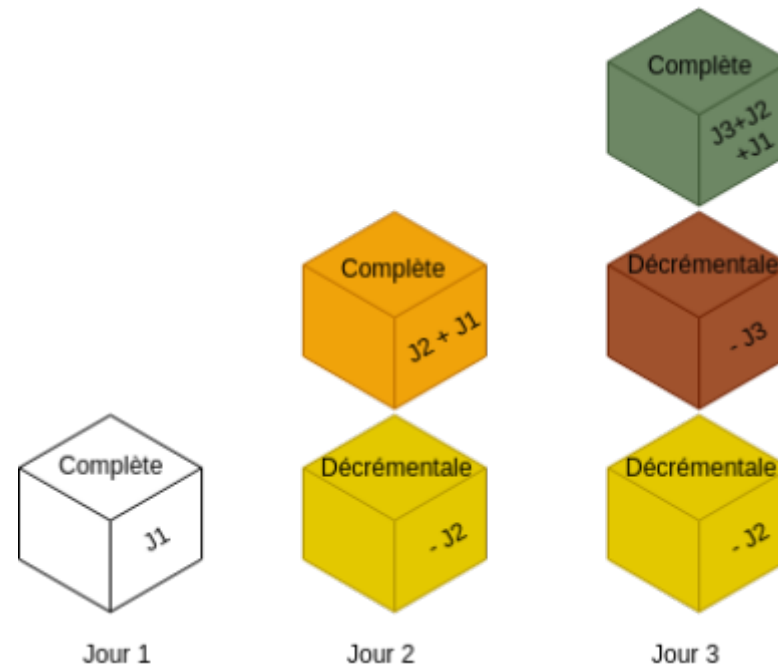
- Sauvegarde plus rapide que la sauvegarde complète ou différentielle,
- Nécessite moins d'espace de stockage que la sauvegarde différentielle.

Inconvénient :

- Plus longue à restaurer.

Sauvegarde décrementale

Dans ce cas on sauvegarde tous les fichiers puis on produit une sauvegarde de ce qui a changé depuis la dernière sauvegarde complète. Cette sauvegarde est appelée une sauvegarde décrementale :



La restauration nécessite :

- La dernière sauvegarde complète.

La restauration de J-1 nécessite :

- La dernière sauvegarde complète,
- La sauvegarde décrementale J.

Par exemple pour obtenir un état du système J1 quand on est à J2, il convient de restaurer la sauvegarde complète de J2 qui est en réalité J2+J1 puis la sauvegarde décrementale -J2. De cette façon et d'une manière mathématique on obtient : $J2 + J1 - J2 = J1$.

Avantage :

- Restauration plus rapide et plus simple que les restaurations à partir de sauvegardes différentielles et incrémentales.

Inconvénient :

- Nécessite plus de manipulation de données à chaque sauvegarde.

Outils Classiques de Sauvegarde

Préparation

Afin de poursuivre, il convient de créer une arborescence à sauvegarder :

```
SUSE12SP1:~ # mkdir -p /test/repY; mkdir /test/repZ
SUSE12SP1:~ # cd /test/repY; touch Y1 Y2 Y3
SUSE12SP1:/test/repY # cd /test/repZ; touch Z1 Z2
SUSE12SP1:/test/repZ # ls -lR /test
/test:
total 0
drwxr-xr-x 1 root root 12 Oct 15 07:10 repY
drwxr-xr-x 1 root root  8 Oct 15 07:10 repZ

/test/repY:
total 0
-rw-r--r-- 1 root root 0 Oct 15 07:10 Y1
-rw-r--r-- 1 root root 0 Oct 15 07:10 Y2
-rw-r--r-- 1 root root 0 Oct 15 07:10 Y3

/test/repZ:
total 0
-rw-r--r-- 1 root root 0 Oct 15 07:10 Z1
-rw-r--r-- 1 root root 0 Oct 15 07:10 Z2
```

La Commande tar

Présentation

Le programme **tar** a été originellement prévu pour sauvegarder sur des bandes magnétiques, d'où son nom issu de **tape archiver**.

La commande **tar** peut sauvegarder vers :

- un fichier spécial, par exemple le nom d'un lecteur de bande,
- un fichier ordinaire sur disque,
- la sortie standard pour être utilisé dans un pipe.

Options de la Commande

Les options de la commande tar sont :

```
SUSE12SP1:/test/repZ # tar --help
Usage: tar [OPTION...] [FILE]...
GNU 'tar' saves many files together into a single tape or disk archive, and can
restore individual files from the archive.
```

Examples:

```
tar -cf archive.tar foo bar  # Create archive.tar from files foo and bar.
tar -tvf archive.tar         # List all files in archive.tar verbosely.
tar -xf archive.tar          # Extract all files from archive.tar.
```

Main operation mode:

-A, --catenate, --concatenate	append tar files to an archive
-c, --create	create a new archive
-d, --diff, --compare	find differences between archive and file system
--delete	delete from the archive (not on mag tapes!)
-r, --append	append files to the end of an archive
-t, --list	list the contents of an archive

--test-label	test the archive volume label and exit
-u, --update	only append files newer than copy in archive
-x, --extract, --get	extract files from an archive

Operation modifiers:

--check-device	check device numbers when creating incremental archives (default)
-g, --listed-incremental=FILE	handle new GNU-format incremental backup
-G, --incremental	handle old GNU-format incremental backup
--ignore-failed-read	do not exit with nonzero on unreadable files
--level=NUMBER	dump level for created listed-incremental archive
-n, --seek	archive is seekable
--no-check-device	do not check device numbers when creating incremental archives
--no-seek	archive is not seekable
--occurrence[=NUMBER]	process only the NUMBERth occurrence of each file in the archive; this option is valid only in conjunction with one of the subcommands --delete, --diff, --extract or --list and when a list of files is given either on the command line or via the -T option; NUMBER defaults to 1
--sparse-version=MAJOR[.MINOR]	set version of the sparse format to use (implies --sparse)
-S, --sparse	handle sparse files efficiently

Overwrite control:

-k, --keep-old-files	don't replace existing files when extracting, treat them as errors
--keep-directory-symlink	preserve existing symlinks to directories when extracting
--keep-newer-files	don't replace existing files that are newer than

	their archive copies
--no-overwrite-dir	preserve metadata of existing directories
--overwrite	overwrite existing files when extracting
--overwrite-dir	overwrite metadata of existing directories when extracting (default)
--recursive-unlink	empty hierarchies prior to extracting directory
--remove-files	remove files after adding them to the archive
--skip-old-files	don't replace existing files when extracting, silently skip over them
-U, --unlink-first	remove each file prior to extracting over it
-W, --verify	attempt to verify the archive after writing it

Select output stream:

--ignore-command-error	ignore exit codes of children
--no-ignore-command-error	treat non-zero exit codes of children as error
-0, --to-stdout	extract files to standard output
--to-command=COMMAND	pipe extracted files to another program

Handling of file attributes:

--atime-preserve[=METHOD]	preserve access times on dumped files, either by restoring the times after reading (METHOD='replace'; default) or by not setting the times in the first place (METHOD='system')
--delay-directory-restore	delay setting modification times and permissions of extracted directories until the end of extraction
--group=NAME	force NAME as group for added files
--mode=CHANGES	force (symbolic) mode CHANGES for added files
--mtime=DATE-OR-FILE	set mtime for added files from DATE-OR-FILE
-m, --touch	don't extract file modified time
--no-delay-directory-restore	

```
cancel the effect of --delay-directory-restore
option
--no-same-owner      extract files as yourself (default for ordinary
                    users)
--no-same-permissions apply the user's umask when extracting permissions
                    from the archive (default for ordinary users)
--numeric-owner      always use numbers for user/group names
--owner=NAME         force NAME as owner for added files
-p, --preserve-permissions, --same-permissions
                    extract information about file permissions
                    (default for superuser)
--preserve           same as both -p and -s
--same-owner         try extracting files with the same ownership as
                    exists in the archive (default for superuser)
-s, --preserve-order, --same-order
                    member arguments are listed in the same order as
                    the files in the archive
```

Handling of extended file attributes:

```
--acls              Enable the POSIX ACLs support
--no-acls           Disable the POSIX ACLs support
--no-selinux        Disable the SELinux context support
--no-xattrs         Disable extended attributes support
--selinux           Enable the SELinux context support
--xattrs            Enable extended attributes support
--xattrs-exclude=MASK specify the exclude pattern for xattr keys
--xattrs-include=MASK specify the include pattern for xattr keys
```

Device selection and switching:

```
-f, --file=ARCHIVE    use archive file or device ARCHIVE
--force-local         archive file is local even if it has a colon
-F, --info-script=NAME, --new-volume-script=NAME
```

```
-L, --tape-length=NUMBER    run script at end of each tape (implies -M)
                             change tape after writing NUMBER x 1024 bytes
-M, --multi-volume          create/list/extract multi-volume archive
    --rmt-command=COMMAND   use given rmt COMMAND instead of rmt
    --rsh-command=COMMAND   use remote COMMAND instead of rsh
    --volno-file=FILE       use/update the volume number in FILE
```

Device blocking:

```
-b, --blocking-factor=BLOCKS  BLOCKS x 512 bytes per record
-B, --read-full-records       reblock as we read (for 4.2BSD pipes)
-i, --ignore-zeros           ignore zeroed blocks in archive (means EOF)
    --record-size=NUMBER     NUMBER of bytes per record, multiple of 512
```

Archive format selection:

```
-H, --format=FORMAT          create archive of the given format
```

FORMAT is one of the following:

```
gnu                GNU tar 1.13.x format
oldgnu             GNU format as per tar <= 1.12
pax                POSIX 1003.1-2001 (pax) format
posix              same as pax
ustar              POSIX 1003.1-1988 (ustar) format
v7                old V7 tar format
```

```
--old-archive, --portability    same as --format=v7
--pax-option=keyword[[:]=value][,keyword[[:]=value]]...
                                control pax keywords
--posix                  same as --format=posix
-V, --label=TEXT          create archive with volume name TEXT; at
                           list/extract time, use TEXT as a globbing pattern
```

for volume name

Compression options:

-a, --auto-compress use archive suffix to determine the compression program
-I, --use-compress-program=PROG filter through PROG (must accept -d)
-j, --bzip2 filter the archive through bzip2
-J, --xz filter the archive through xz
 --lzip filter the archive through lzip
 --lzma filter the archive through lzma
 --lzop
 --no-auto-compress do not use archive suffix to determine the compression program
-z, --gzip, --gunzip, --ungzip filter the archive through gzip
-Z, --compress, --uncompress filter the archive through compress

Local file selection:

 --add-file=FILE add given FILE to the archive (useful if its name starts with a dash)
 --backup[=CONTROL] backup before removal, choose version CONTROL
-C, --directory=DIR change to directory DIR
 --exclude=PATTERN exclude files, given as a PATTERN
 --exclude-backups exclude backup and lock files
 --exclude-caches exclude contents of directories containing CACHEDIR.TAG, except for the tag file itself
 --exclude-caches-all exclude directories containing CACHEDIR.TAG
 --exclude-caches-under exclude everything under directories containing CACHEDIR.TAG
 --exclude-tag=FILE exclude contents of directories containing FILE, except for FILE itself
 --exclude-tag-all=FILE exclude directories containing FILE

```
--exclude-tag-under=FILE    exclude everything under directories
                             containing FILE
--exclude-vcs                exclude version control system directories
-h, --dereference            follow symlinks; archive and dump the files they
                             point to
--hard-dereference           follow hard links; archive and dump the files they
                             refer to
-K, --starting-file=MEMBER-NAME
                             begin at member MEMBER-NAME when reading the
                             archive
--newer-mtime=DATE           compare date and time when data changed only
--no-null                    disable the effect of the previous --null option
--no-recursion               avoid descending automatically in directories
--no-unquote                 do not unquote filenames read with -T
--null                       -T reads null-terminated names, disable -C
-N, --newer=DATE-OR-FILE, --after-date=DATE-OR-FILE
                             only store files newer than DATE-OR-FILE
--one-file-system            stay in local file system when creating archive
-P, --absolute-names         don't strip leading '/'s from file names
--recursion                  recurse into directories (default)
--suffix=STRING              backup before removal, override usual suffix ('~'
                             unless overridden by environment variable
                             SIMPLE_BACKUP_SUFFIX)
-T, --files-from=FILE        get names to extract or create from FILE
--unquote                    unquote filenames read with -T (default)
-X, --exclude-from=FILE      exclude patterns listed in FILE
```

File name transformations:

```
--strip-components=NUMBER    strip NUMBER leading components from file
                             names on extraction
--transform=EXPRESSION, --xform=EXPRESSION
                             use sed replace EXPRESSION to transform file
                             names
```

File name matching options (affect both exclude and include patterns):

- anchored patterns match file name start
- ignore-case ignore case
- no-anchored patterns match after any '/' (default for exclusion)
- no-ignore-case case sensitive matching (default)
- no-wildcards verbatim string matching
- no-wildcards-match-slash wildcards do not match '/'
- wildcards use wildcards (default for exclusion)
- wildcards-match-slash wildcards match '/' (default for exclusion)

Informative output:

- checkpoint[=NUMBER] display progress messages every NUMBERth record (default 10)
- checkpoint-action=ACTION execute ACTION on each checkpoint
- full-time print file time to its full resolution
- index-file=FILE send verbose output to FILE
- l, --check-links print a message if not all links are dumped
- no-quote-chars=STRING disable quoting for characters from STRING
- quote-chars=STRING additionally quote characters from STRING
- quoting-style=STYLE set name quoting style; see below for valid STYLE values
- R, --block-number show block number within archive with each message
- show-defaults show tar defaults
- show-omitted-dirs when listing or extracting, list each directory that does not match search criteria
- show-snapshot-field-ranges show valid ranges for snapshot-file fields
- show-transformed-names, --show-stored-names show file or archive names after transformation
- totals[=SIGNAL] print total bytes after processing the archive; with an argument - print total bytes when this

	SIGNAL is delivered; Allowed signals are: SIGHUP, SIGQUIT, SIGINT, SIGUSR1 and SIGUSR2; the names without SIG prefix are also accepted
--utc	print file modification times in UTC
-v, --verbose	verbosely list files processed
--warning=KEYWORD	warning control
-w, --interactive, --confirmation	ask for confirmation for every action

Compatibility options:

-o	when creating, same as --old-archive; when extracting, same as --no-same-owner
----	--

Other options:

-, --help	give this help list
--restrict	disable use of some potentially harmful options
--usage	give a short usage message
--version	print program version

Mandatory or optional arguments to long options are also mandatory or optional for any corresponding short options.

The backup suffix is '~', unless set with --suffix or SIMPLE_BACKUP_SUFFIX.
The version control may be set with --backup or VERSION_CONTROL, values are:

none, off	never make backups
t, numbered	make numbered backups
nil, existing	numbered if numbered backups exist, simple otherwise
never, simple	always make simple backups

Valid arguments for the --quoting-style option are:

```
literal
shell
shell-always
c
c-maybe
escape
locale
clocale
```

This tar defaults to:

```
--format=posix -f- -b20 --quoting-style=escape --rmt-command=/usr/lib/rmt
--rsh-command=/usr/bin/rsh
```

Report bugs to <bug-tar@gnu.org>.

LAB #1 - Travailler avec la Commande tar

Vous allez maintenant sauvegarder votre dossier **test** ainsi que son contenu vers un fichier :

```
SUSE12SP1:/test/repZ # tar cvf /tmp/test.tar /test
tar: Removing leading `/' from member names
/test/
/test/repY/
/test/repY/Y1
/test/repY/Y2
/test/repY/Y3
/test/repZ/
/test/repZ/Z1
/test/repZ/Z2
```

Pour visualiser la **table of contents** de votre sauvegarde, utilisez la commande suivante :

```
SUSE12SP1:/test/repZ # tar tvf /tmp/test.tar
drwxr-xr-x root/root      0 2017-10-15 07:10 test/
drwxr-xr-x root/root      0 2017-10-15 07:10 test/repY/
-rw-r--r-- root/root      0 2017-10-15 07:10 test/repY/Y1
-rw-r--r-- root/root      0 2017-10-15 07:10 test/repY/Y2
-rw-r--r-- root/root      0 2017-10-15 07:10 test/repY/Y3
drwxr-xr-x root/root      0 2017-10-15 07:10 test/repZ/
-rw-r--r-- root/root      0 2017-10-15 07:10 test/repZ/Z1
-rw-r--r-- root/root      0 2017-10-15 07:10 test/repZ/Z2
```

Afin de créer une sauvegarde incrémentale, vous avez besoin de créer un fichier qui servira de référence de date :

```
SUSE12SP1:/test/repZ # touch /tmp/dateref
```

Modifiez maintenant deux des fichiers de votre arborescence **test** :

```
SUSE12SP1:/test/repZ # echo "SLES est super \!" > /test/repY/Y1
SUSE12SP1:/test/repZ # echo "SLES is wonderful \!" > /test/repZ/Z1
```

Pour procéder à votre sauvegarde incrémentale, vous devez sauvegarder uniquement les fichiers modifiés ou créés depuis la création de votre fichier **/tmp/dateref**.

Saisissez donc la commande suivante :

```
SUSE12SP1:/test/repZ # tar -cvf /tmp/incremental.tar -N /tmp/dateref /test
tar: Removing leading `/' from member names
/test/
/test/repY/
/test/repY/Y1
tar: /test/repY/Y2: file is unchanged; not dumped
tar: /test/repY/Y3: file is unchanged; not dumped
/test/repZ/
/test/repZ/Z1
```

```
tar: /test/repZ/Z2: file is unchanged; not dumped
```



Important - Notez l'utilisation de l'option **-N** avec l'argument **/tmp/dateref** qui permet d'identifier les fichiers modifiés ou créés depuis la création de **/tmp/dateref**.

Contrôlez maintenant le contenu de l'archive **/tmp/incremental.tar** :

```
SUSE12SP1:/test/repZ # tar tvf /tmp/incremental.tar
drwxr-xr-x root/root      0 2017-10-15 07:10 test/
drwxr-xr-x root/root      0 2017-10-15 07:10 test/repY/
-rw-r--r-- root/root    18 2017-10-15 08:07 test/repY/Y1
drwxr-xr-x root/root      0 2017-10-15 07:10 test/repZ/
-rw-r--r-- root/root    21 2017-10-15 08:07 test/repZ/Z1
```

Supprimez maintenant le contenu du répertoire **test** :

```
SUSE12SP1:/test/repZ # rm -rf /test/*
```



Important - Notez que le système vous permet de supprimer le répertoire **/test/repZ**, or vous vous situez dans ce même répertoire !

Afin de pouvoir restaurer les fichiers de votre première sauvegarde, placez-vous à la racine de votre système et restaurez le contenu de votre répertoire **test** en saisissant la commande tar suivante :

```
SUSE12SP1:/test/repZ # cd /
SUSE12SP1:/ # tar xvf /tmp/test.tar
test/
test/repY/
test/repY/Y1
```

```
test/repY/Y2
test/repY/Y3
test/repZ/
test/repZ/Z1
test/repZ/Z2
```

Constatez maintenant que l'opération s'est bien déroulée :

```
SUSE12SP1:/ # ls -lR /test
/test:
total 0
drwxr-xr-x 1 root root 12 Oct 15 07:10 repY
drwxr-xr-x 1 root root  8 Oct 15 07:10 repZ

/test/repY:
total 0
-rw-r--r-- 1 root root 0 Oct 15 07:10 Y1
-rw-r--r-- 1 root root 0 Oct 15 07:10 Y2
-rw-r--r-- 1 root root 0 Oct 15 07:10 Y3

/test/repZ:
total 0
-rw-r--r-- 1 root root 0 Oct 15 07:10 Z1
-rw-r--r-- 1 root root 0 Oct 15 07:10 Z2
```



Important - Notez qu'à ce stade les fichiers **/test/repY/Y1** et **/test/repZ/Z1** sont vides.

Restaurez maintenant votre archive incrémentale :

```
SUSE12SP1:/ # tar xvf /tmp/incremental.tar
test/
```

```
test/repY/  
test/repY/Y1  
test/repZ/  
test/repZ/Z1
```

Constatez maintenant que l'opération s'est bien déroulée :

```
SUSE12SP1:/ # ls -lR /test  
/test:  
total 0  
drwxr-xr-x 1 root root 12 Oct 15 07:10 repY  
drwxr-xr-x 1 root root  8 Oct 15 07:10 repZ  
  
/test/repY:  
total 4  
-rw-r--r-- 1 root root 18 Oct 15 08:07 Y1  
-rw-r--r-- 1 root root  0 Oct 15 07:10 Y2  
-rw-r--r-- 1 root root  0 Oct 15 07:10 Y3  
  
/test/repZ:  
total 4  
-rw-r--r-- 1 root root 21 Oct 15 08:07 Z1  
-rw-r--r-- 1 root root  0 Oct 15 07:10 Z2
```



Important - Notez que les fichiers **/test/repY/Y1** et **/test/repZ/Z1** sont maintenant non-vides.

La Commande GPL tar et la Compression

Dernièrement, la commande tar peut archiver en utilisant des algorithmes de compression :

Algorithme	Option de la commande tar
gzip	z
bzip2	j
lzma	J

La Commande cpio

Présentation

La commande **cpio** (Copy Input To Output). cpio peut gérer les archives au format **tar**. La différence majeure entre tar et cpio est que ce dernier stocke les chemins d'accès aux fichiers sauvegardés en même temps que les fichiers eux-mêmes. Ceci implique que dans le cas où le chemin absolu a été spécifié lors de la sauvegarde, il est impossible de restaurer un fichier à un autre emplacement que son emplacement d'origine.

Vous allez utiliser maintenant le logiciel **cpio** pour effectuer les sauvegardes et restaurations.

Options de la Commande

Les options de la commande **cpio** sont :

```
SUSE12SP1:/ # cpio --help
Usage: cpio [OPTION...] [destination-directory]
GNU `cpio' copies files to and from archives

Examples:
# Copy files named in name-list to the archive
cpio -o < name-list [> archive]
# Extract files from the archive
cpio -i [< archive]
# Copy files named in name-list to destination-directory
cpio -p destination-directory < name-list
```

Main operation mode:

- i, --extract Extract files from an archive (run in copy-in mode)
- o, --create Create the archive (run in copy-out mode)
- p, --pass-through Run in copy-pass mode
- t, --list Print a table of contents of the input

Operation modifiers valid in any mode:

- block-size=BLOCK-SIZE Set the I/O block size to BLOCK-SIZE * 512 bytes
- B Set the I/O block size to 5120 bytes
- c Use the old portable (ASCII) archive format
- C, --io-size=NUMBER Set the I/O block size to the given NUMBER of bytes
- force-local Archive file is local, even if its name contains colons
- f, --nonmatching Only copy files that do not match any of the given patterns
- F, --file=[[USER@]HOST:]FILE-NAME
 Use this FILE-NAME instead of standard input or output. Optional USER and HOST specify the user and host names in case of a remote archive
- H, --format=FORMAT Use given archive FORMAT
- M, --message=STRING Print STRING when the end of a volume of the backup media is reached
- n, --numeric-uid-gid In the verbose table of contents listing, show numeric UID and GID
- quiet Do not print the number of blocks copied
- rsh-command=COMMAND Use remote COMMAND instead of rsh
- v, --verbose Verbosely list the files processed
- V, --dot Print a "." for each file processed
- W, --warning=FLAG Control warning display. Currently FLAG is one of 'none', 'truncate', 'all'. Multiple options

accumulate.

Operation modifiers valid only in copy-in mode:

-b, --swap	Swap both halfwords of words and bytes of halfwords in the data. Equivalent to -sS
-r, --rename	Interactively rename files
-s, --swap-bytes	Swap the bytes of each halfword in the files
-S, --swap-halfwords	Swap the halfwords of each word (4 bytes) in the files
--to-stdout	Extract files to standard output
-E, --pattern-file=FILE	Read additional patterns specifying filenames to extract or list from FILE
--only-verify-crc	When reading a CRC format archive, only verify the CRC's of each file in the archive, don't actually extract the files

Operation modifiers valid only in copy-out mode:

-A, --append	Append to an existing archive.
-O [[USER@]HOST:]FILE-NAME	Archive filename to use instead of standard output. Optional USER and HOST specify the user and host names in case of a remote archive

Operation modifiers valid only in copy-pass mode:

-l, --link	Link files instead of copying them, when possible
------------	---

Operation modifiers valid in copy-in and copy-out modes:

--absolute-filenames	Do not strip file system prefix components from the file names
----------------------	--

--no-absolute-filenames Create all files relative to the current directory

Operation modifiers valid in copy-out and copy-pass modes:

-0, --null A list of filenames is terminated by a null character instead of a newline

-a, --reset-access-time Reset the access times of files after reading them

-I [[USER@]HOST:]FILE-NAME Archive filename to use instead of standard input. Optional USER and HOST specify the user and host names in case of a remote archive

-L, --dereference Dereference symbolic links (copy the files that they point to instead of copying the links).

-R, --owner=[USER][:.][GROUP] Set the ownership of all files created to the specified USER and/or GROUP

Operation modifiers valid in copy-in and copy-pass modes:

-d, --make-directories Create leading directories where needed

--extract-over-symlinks Force writing over symbolic links

-m, --preserve-modification-time Retain previous file modification times when creating files

--no-preserve-owner Do not change the ownership of the files

--sparse Write files with large blocks of zeros as sparse files

-u, --unconditional Replace all files unconditionally

-, --help give this help list

--usage give a short usage message

--version print program version

Mandatory or optional arguments to long options are also mandatory or optional for any corresponding short options.

Report bugs to <bug-cpio@gnu.org>.

LAB #2 - Travailler avec la Commande cpio

Dans un premier temps, vous devez utiliser la commande **find** pour construire une liste de fichiers à sauvegarder :

```
SUSE12SP1:/ # find /test > /tmp/cpio.liste
SUSE12SP1:/ # cat /tmp/cpio.liste
/test
/test/repY
/test/repY/Y2
/test/repY/Y3
/test/repY/Y1
/test/repZ
/test/repZ/Z2
/test/repZ/Z1
```

Sauvegardez maintenant les fichiers et répertoires référencés par le fichier **/tmp/cpio.liste** :

```
SUSE12SP1:/ # cpio -ov < /tmp/cpio.liste > /tmp/test.cpio
/test
/test/repY
/test/repY/Y2
/test/repY/Y3
/test/repY/Y1
/test/repZ
/test/repZ/Z2
/test/repZ/Z1
1 block
```

Consultez maintenant la **table of contents** de votre sauvegarde :

```
SUSE12SP1:/ # cpio -it < /tmp/test.cpio
/test
/test/repY
/test/repY/Y2
/test/repY/Y3
/test/repY/Y1
/test/repZ
/test/repZ/Z2
/test/repZ/Z1
1 block
```

Supprimez maintenant le répertoire **/test/repY** et son contenu :

```
SUSE12SP1:/ # rm -rf /test/repY
```

Contrôlez le bon déroulement de la suppression :

```
SUSE12SP1:/ # ls -lR /test
/test:
total 0
drwxr-xr-x 1 root root 8 Oct 15 07:10 repZ

/test/repZ:
total 4
-rw-r--r-- 1 root root 21 Oct 15 08:07 Z1
-rw-r--r-- 1 root root  0 Oct 15 07:10 Z2
```

Restaurez les fichiers supprimés :

```
SUSE12SP1:/ # cpio -ivdum "/test/repY/*" < /tmp/test.cpio
/test/repY/Y2
/test/repY/Y3
/test/repY/Y1
```

1 block



Important - Notez l'utilisation de la chaîne **“/test/repY/*”** qui permet de rechercher uniquement le répertoire **repY** ainsi que les fichiers **Y1, Y2** et **Y3** dans l'archive test.cpio.

Les options utilisées avec la commande cpio sont :



1. i, -extract Extract files from an archive (run in copy-in mode)
2. v, -verbose Verbosely list the files processed
3. d, -make-directories Create leading directories where needed
4. u, -unconditional Replace all files unconditionally
5. m, -preserve-modification-time

Retain previous file modification times when creating files

Contrôlez le bon déroulement de la restauration :

```
SUSE12SP1:/ # ls -lR /test
/test:
total 0
drwx----- 1 root root 12 Oct 15 08:15 repY
drwxr-xr-x 1 root root  8 Oct 15 07:10 repZ

/test/repY:
total 4
-rw-r--r-- 1 root root 18 Oct 15 08:07 Y1
-rw-r--r-- 1 root root  0 Oct 15 07:10 Y2
-rw-r--r-- 1 root root  0 Oct 15 07:10 Y3
```

```
/test/repZ:
total 4
-rw-r--r-- 1 root root 21 Oct 15 08:07 Z1
-rw-r--r-- 1 root root  0 Oct 15 07:10 Z2
```

La Commande dd

Présentation

La commande **dd** n'est pas réellement une commande de sauvegarde.

La commande **dd** copie le fichier passé en entrée dans le fichier de sortie en limitant le nombre d'octets copiés par l'utilisation de deux options :

- **count**
 - le nombre
- **bs**
 - la taille du bloc à copier

Options de la Commande

Les options de la commande **dd** sont :

```
SUSE12SP1:/ # dd --help
Usage: dd [OPERAND]...
       or: dd OPTION
Copy a file, converting and formatting according to the operands.

bs=BYTES      read and write up to BYTES bytes at a time
cbs=BYTES      convert BYTES bytes at a time
conv=CONVS     convert the file as per the comma separated symbol list
count=N        copy only N input blocks
```

ibs=BYTES	read up to BYTES bytes at a time (default: 512)
if=FILE	read from FILE instead of stdin
iflag=FLAGS	read as per the comma separated symbol list
obs=BYTES	write BYTES bytes at a time (default: 512)
of=FILE	write to FILE instead of stdout
oflag=FLAGS	write as per the comma separated symbol list
seek=N	skip N obs-sized blocks at start of output
skip=N	skip N ibs-sized blocks at start of input
status=WHICH	WHICH info to suppress outputting to stderr; 'noxfers' suppresses transfer stats, 'none' suppresses all

N and BYTES may be followed by the following multiplicative suffixes:

c =1, w =2, b =512, kB =1000, K =1024, MB =1000*1000, M =1024*1024, xM =M
GB =1000*1000*1000, G =1024*1024*1024, and so on for T, P, E, Z, Y.

Each CONV symbol may be:

ascii	from EBCDIC to ASCII
ebcdic	from ASCII to EBCDIC
ibm	from ASCII to alternate EBCDIC
block	pad newline-terminated records with spaces to cbs-size
unblock	replace trailing spaces in cbs-size records with newline
lcase	change upper case to lower case
ucase	change lower case to upper case
sparse	try to seek rather than write the output for NUL input blocks
swab	swap every pair of input bytes
sync	pad every input block with NULs to ibs-size; when used with block or unblock, pad with spaces rather than NULs
excl	fail if the output file already exists
nocreat	do not create the output file
notrunc	do not truncate the output file
noerror	continue after read errors
fdatsync	physically write output file data before finishing
fsync	likewise, but also write metadata

Each FLAG symbol may be:

append	append mode (makes sense only for output; conv=notrunc suggested)
direct	use direct I/O for data
directory	fail unless a directory
dsync	use synchronized I/O for data
sync	likewise, but also for metadata
fullblock	accumulate full blocks of input (iflag only)
nonblock	use non-blocking I/O
noatime	do not update access time
nocache	discard cached data
noctty	do not assign controlling terminal from file
nofollow	do not follow symlinks
count_bytes	treat 'count=N' as a byte count (iflag only)
skip_bytes	treat 'skip=N' as a byte count (iflag only)
seek_bytes	treat 'seek=N' as a byte count (oflag only)

Sending a USR1 signal to a running 'dd' process makes it print I/O statistics to standard error and then resume copying.

```
$ dd if=/dev/zero of=/dev/null& pid=$!  
$ kill -USR1 $pid; sleep 1; kill $pid  
18335302+0 records in  
18335302+0 records out  
9387674624 bytes (9.4 GB) copied, 34.6279 seconds, 271 MB/s
```

Options are:

--help	display this help and exit
--version	output version information and exit

GNU coreutils online help: <<http://www.gnu.org/software/coreutils/>>
Report dd translation bugs to <<http://translationproject.org/team/>>
For complete documentation, run: info '(coreutils) dd invocation'

LAB #3 - Travailler avec la Commande dd

Vous allez utiliser maintenant le logiciel **dd** pour effectuer une sauvegarde de votre MBR et de la table des partitions.

Effectuez une sauvegarde de votre MBR qui se trouve dans les premiers 446 octets de votre disque **/dev/sda** :

```
SUSE12SP1:/ # dd if=/dev/sda of=/tmp/mbr.save bs=1 count=446
446+0 records in
446+0 records out
446 bytes (446 B) copied, 0.00915145 s, 48.7 kB/s
```

Effectuez maintenant une sauvegarde de votre table des partitions qui se trouve dans les 64 octets après les 446 précédemment sauvegardés :

```
SUSE12SP1:/ # dd if=/dev/sda of=/tmp/tblpart.save bs=1 count=64 skip=446
64+0 records in
64+0 records out
64 bytes (64 B) copied, 0.00173448 s, 36.9 kB/s
```



Important - Notez l'utilisation de l'option **skip** qui permet de positionner le début de la sauvegarde au 447^{ième} octet.

dump et restore

Présentation

Les commandes **dump** et **restore** se basent sur le format d'enregistrement des données (ext3). Pour cette raison il n'est pas possible de sauvegarder des répertoires à l'intérieur d'un système de fichiers mais uniquement des systèmes de fichiers complets.

Il est important de noter que le système de fichier ne doit pas être utilisé pendant le processus de dump. Pour cette raison il est normalement conseillé de démonter le système de fichiers.

Il existe 10 niveaux de dump possibles de **0** à **9**. Lors d'un dump le niveau est spécifié. Chaque fois qu'un dump est effectué, cette information est sauvegardée dans le fichier `/etc/dumpdates`.

Par définition un dump de niveau **0** est une sauvegarde complète tandis que le dump de niveau 1 est une sauvegarde incrémentale.

Notez que les fichiers sont sauvegardés avec des nom relatifs. Ceci implique que vous devez vous positionner dans le système de fichiers lors de la restauration avec la commande **restore**.

Outils Avancés de Sauvegarde

Outils de Sauvegarde Unidirectionnelle

Ces outils sauvegardent des fichiers vers un répertoire local ou distant dans un seul sens.

Ce premier tableau compare les outils par rapport aux caractéristiques et aux capacités des opérations de sauvegarde :

Outil	Backend	Sauvegarde différentielle	Sauvegarde incrémentale	Sauvegarde décrementale	Planification incorporée	Restauration incorporée	Chiffrement	Compression	Site
AMANDA	tar, gzip								AMANDA
Areca Backup	Aucun								Areca Backup
bacula (Bareos)	MySQL								bacula
BackInTime	rsync, diff, meld								BackInTime
BackupPC	rsync, samba, tar								BackupPC
Dar	Aucun								Dar

Outil	Backend	Sauvegarde différentielle	Sauvegarde incrémentale	Sauvegarde décrementale	Planification incorporée	Restauration incorporée	Chiffrement	Compression	Site
Déjà Dup	duplicity	✗	✓	✗	✓	✓	✓	✗	Duplicity
Grsync	rsync	✗	✓	✗	✓	✓	✗	✗	Grsync
luckyBackup	rsync	✗	✓	✗	✓	✓	✗	✗	luckyBackup
SBackup	Aucun	✗	✓	✗	✓	✓	✗	✓	SBackup



A Faire - Consultez la page [List of backup software](#) sur Wikipedia pour plus d'informations.

Ce deuxième tableau compare les outils triés par la date de dernière version connue :

Outil	Ecrit en	Licence	Dernière Version	Date Version	Linux	Windows	Mac	Site Web
Dar	C++	GPL	2.6.2	09/02/2019	✓	✓	✓	Dar
Bacula (Bareos)	C, C++	GNU Affero General Public License v3.0	9.4.2	04/02/2019	✓	✓	✓	bacula
BackupPC	Perl	GPL v3.0	4.3.0	25/11/2018	✓	✓	✓	BackupPC
luckyBackup	C++	GPL v3.0	0.5.0	18/11/2018	✓	✓	✓	luckyBackup
Déjà Dup (Duplicity)	Python	GPL	0.7.18.2	17/11/2018	✓	✓	✓	Duplicity
AMANDA	C, Perl	GPL, LGPL, Apache, Amanda License	3.5.1	01/12/2017	✓	✓	✓	AMANDA
BackInTime	Python3	GPL	1.1.24	07/11/2017	✓	✗	✗	BackInTime
Grsync	GTK	GPL	1.2.6	15/03/2016	✓	✓	✓	Grsync

Outil	Ecrit en	Licence	Dernière Version	Date Version	Linux	Windows	Mac	Site Web
Areca Backup	Java	GPLv2	7.5	26/08/2015	✓	✓	✓	Areca Backup
SBackup	GTK	GPLv2	0.11.6	24/02/2014	✓	✗	✗	SBackup

Outils de Sauvegarde Multidirectionnelle

Ces outils synchronisent les fichiers entre deux serveurs.

Ce premier tableau compare les outils par rapport aux caractéristiques et aux capacités des opérations de sauvegarde :

Outil	Backend	Sauvegarde différentielle	Sauvegarde incrémentale	Sauvegarde décrementale	Planification incorporée	Restauration incorporée	Chiffrement	Compression
FullSync	smb, ftp, sftp	✗	✓	✗	✓	S/O	✗	✗
FreeFileSync	Aucun	✗	✓	✗	✓	S/O	✗	✗
unison	SSH, RSH	✗	✓	✗	✓	S/O	✗	✗
Synkron	Aucun	✗	✓	✗	✓	✓	✗	✗



A Faire - Consultez la page [Comparison of file synchronization software](#) sur Wikipedia pour plus d'informations..

Ce deuxième tableau compare les outils triés par la date de dernière version connue :

Outil	Ecrit en	Licence	Dernière Version	Date Version	Linux	Windows	Mac	Site Web
FreeFileSync	Divers	GPL v3.0	10.8	05/01/2019	✓	✓	✓	FreeFileSync
unison	OCaml	GPL v3.0	2.51.2	27/01/2018	✓	✓	✓	unison

Outil	Ecrit en	Licence	Dernière Version	Date Version	Linux	Windows	Mac	Site Web
FullSync	Java	GPLv2	0.10.4	05/04/2016	✓	✓	✗	FullSync
Synkron	C++	GPL v2	1.6.2	25/01/2011	✓	✓	✓	Synkron

Outils de Sauvegarde des Partitions

Ce premier tableau compare les outils par rapport aux caractéristiques et aux capacités des opérations de sauvegarde :

Outil	Backend	Sauvegarde différentielle	Sauvegarde incrémentale	Sauvegarde décrémentationale	Planification incorporée	Restauration incorporée	Chiffrement	Compression
CloneZilla	SSH, samba, NFS	✗	✗	✗	✗	✓	✓	✓
Partclone	Aucun	✗	✗	✗	✗	✓	✗	✓
partimage	Aucun	✗	✗	✗	✗	✓	✗	✗



A Faire - Consultez la page [Comparison of disk cloning software](#) sur Wikipedia pour plus d'informations.

Ce deuxième tableau compare les outils triés par la date de dernière version connue :

Outil	Ecrit en	Licence	Dernière Version	Date Version	Linux	Windows	Mac	Site Web
CloneZilla	Perl, Unix shell	GPL	2.6.0-37	10/01/2019	✓	✓	✓	CloneZilla
Partclone	C	GPL	0.2.89	05/07/2016	✓	✓	✓	Partclone
partimage	C	GPL	0.6.9	25/07/2010	✓	✓	✗	Partimage

Compression

La Commande gzip

Présentation

La commande **gzip** est un utilitaire de compression sous GNU/Linux. La commande **gunzip** est un utilitaire de décompression sous GNU/Linux.

Options des Commandes

Les options de la commande **gzip** sont :

```
SUSE12SP1:/ # gzip --help
Usage: gzip [OPTION]... [FILE]...
Compress or uncompress FILEs (by default, compress FILES in-place).

Mandatory arguments to long options are mandatory for short options too.

  -c, --stdout      write on standard output, keep original files unchanged
  -d, --decompress  decompress
  -f, --force       force overwrite of output file and compress links
  -h, --help        give this help
  -k, --keep        keep (don't delete) input files
  -l, --list        list compressed file contents
  -L, --license     display software license
  -n, --no-name     do not save or restore the original name and time stamp
  -N, --name        save or restore the original name and time stamp
  -q, --quiet       suppress all warnings
  -r, --recursive  operate recursively on directories
  -S, --suffix=SUF use suffix SUF on compressed files
```

```
-t, --test      test compressed file integrity
-v, --verbose   verbose mode
-V, --version   display version number
-l, --fast      compress faster
-9, --best      compress better
    --rsyncable  Make rsync-friendly archive
```

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>.

Les options de la commande **gunzip** sont :

```
SUSE12SP1:/ # gunzip --help
Usage: /usr/bin/gunzip [OPTION]... [FILE]...
Uncompress FILEs (by default, in-place).
```

Mandatory arguments to long options are mandatory for short options too.

```
-c, --stdout    write on standard output, keep original files unchanged
-f, --force     force overwrite of output file and compress links
-k, --keep      keep (don't delete) input files
-l, --list      list compressed file contents
-n, --no-name   do not save or restore the original name and time stamp
-N, --name      save or restore the original name and time stamp
-q, --quiet     suppress all warnings
-r, --recursive operate recursively on directories
-S, --suffix=SUF use suffix SUF on compressed files
-t, --test      test compressed file integrity
-v, --verbose   verbose mode
    --help      display this help and exit
    --version   display version information and exit
```

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>.

LAB #4 - Travailler avec la Commande gzip

Utilisez **gzip** pour compresser votre fichier tar :

```
SUSE12SP1:/ # gzip /tmp/test.tar
```

Constatez la taille du fichier **test.tar.gz** :

```
SUSE12SP1:/ # ls -l /tmp/test.tar.gz  
-rw-r--r-- 1 root root 497 Oct 15 08:05 /tmp/test.tar.gz
```



Important - Notez que le fichier compressé a été créé dans le même répertoire que le fichier source et que le fichier source a disparu.

Décompressez le fichier test.tar.gz :

```
SUSE12SP1:/ # gunzip /tmp/test.tar.gz
```

La Commande bzip2

Présentation

La commande **bzip2** est un utilitaire de compression sous GNU/Linux. La commande **bunzip2** est un utilitaire de décompression sous GNU/Linux.

Options des Commandes

Les options de la commande **bzip2** sont :

```
SUSE12SP1:/ # bzip2 --help
bzip2, a block-sorting file compressor.  Version 1.0.6, 6-Sept-2010.
```

```
usage: bzip2 [flags and input files in any order]
```

```
-h --help           print this message
-d --decompress     force decompression
-z --compress       force compression
-k --keep           keep (don't delete) input files
-f --force          overwrite existing output files
-t --test           test compressed file integrity
-c --stdout         output to standard out
-q --quiet          suppress noncritical error messages
-v --verbose        be verbose (a 2nd -v gives more)
-L --license        display software version & license
-V --version        display software version & license
-s --small          use less memory (at most 2500k)
-1 .. -9           set block size to 100k .. 900k
--fast             alias for -1
--best             alias for -9
```

```
If invoked as `bzip2', default action is to compress.
      as `bunzip2', default action is to decompress.
      as `bzcata', default action is to decompress to stdout.
```

```
If no file names are given, bzip2 compresses or decompresses
from standard input to standard output.  You can combine
short flags, so `-v -4' means the same as -v4 or -4v, &c.
```

Les options de la commande **bunzip2** sont :

```
SUSE12SP1:/ # bunzip2 --help
bzip2, a block-sorting file compressor.  Version 1.0.6, 6-Sept-2010.
```

```
usage: bunzip2 [flags and input files in any order]
```

```
-h --help          print this message
-d --decompress    force decompression
-z --compress      force compression
-k --keep          keep (don't delete) input files
-f --force         overwrite existing output files
-t --test          test compressed file integrity
-c --stdout        output to standard out
-q --quiet         suppress noncritical error messages
-v --verbose       be verbose (a 2nd -v gives more)
-L --license       display software version & license
-V --version       display software version & license
-s --small         use less memory (at most 2500k)
-l .. -9          set block size to 100k .. 900k
--fast            alias for -l
--best            alias for -9
```

```
If invoked as `bzip2', default action is to compress.
      as `bunzip2', default action is to decompress.
      as `bzipcat', default action is to decompress to stdout.
```

```
If no file names are given, bzip2 compresses or decompresses
from standard input to standard output.  You can combine
short flags, so `-v -4' means the same as -v4 or -4v, &c.
```

LAB #5 - Travailler avec la Commande bzip2

Utilisez **bzip2** pour compresser votre fichier tar :

```
SUSE12SP1:/ # bzip2 /tmp/test.tar
```

Constatez la taille du fichier **tar.bz2** :

```
SUSE12SP1:/ # ls -l /tmp | grep test.tar.bz2  
-rw-r--r-- 1 root    root    419 Oct 15 08:05 test.tar.bz2
```



Important - Notez que le fichier compressé a été créé dans le même répertoire que le fichier source et que le fichier source a disparu.

Décompressez le fichier tar.bz2 :

```
SUSE12SP1:/ # bunzip2 /tmp/test.tar.bz2
```

La Commande xz

Présentation

La commande **xz** est un utilitaire de compression sous GNU/Linux. D'autres commandes sont :

- **unxz** - équivalent à **xz -decompress**.
- **xzcat** - équivalent à **xz -decompress -stdout**.
- **lzma** - équivalent à **xz -format=lzma**.
- **unlzma** - équivalent à **xz -format=lzma -decompress**.
- **lzcat** - équivalent à **xz -format=lzma -decompress -stdout**.

La commande xz ne compressera pas le fichier si :

- le fichier n'est pas de type standard
- le fichier est un lien symbolique
- le fichier est un lien physique
- le fichier possède le sticky bit, le SUID bit ou le SGID bit
- le fichier possède déjà une extension .xz ou .lzma

La commande xz ne décompressera pas le fichier si :

- le fichier ne possède pas d'extension .xz ou .lzma

Options des Commandes

Les options de la commande **xz** sont :

```
SUSE12SP1:/ # xz --help
Usage: xz [OPTION]... [FILE]...
Compress or decompress FILEs in the .xz format.

  -z, --compress      force compression
  -d, --decompress    force decompression
  -t, --test          test compressed file integrity
  -l, --list          list information about .xz files
  -k, --keep          keep (don't delete) input files
  -f, --force         force overwrite of output file and (de)compress links
  -c, --stdout        write to standard output and don't delete input files
  -0 ... -9          compression preset; default is 6; take compressor *and*
                    decompressor memory usage into account before using 7-9!
  -e, --extreme       try to improve compression ratio by using more CPU time;
                    does not affect decompressor memory requirements
  -q, --quiet         suppress warnings; specify twice to suppress errors too
  -v, --verbose       be verbose; specify twice for even more verbose
```

```
-h, --help      display this short help and exit
-H, --long-help display the long help (lists also the advanced options)
-V, --version   display the version number and exit
```

With no FILE, or when FILE is -, read standard input.

Report bugs to <lasse.collin@tukaani.org> (in English or Finnish).
XZ Utils home page: <<http://tukaani.org/xz/>>

LAB #6 - Travailler avec la Commande xz

Utilisez **xz** pour compresser votre fichier tar :

```
SUSE12SP1:/ # xz /tmp/test.tar
```



Important - Notez que le fonctionnement par défaut de la commande est identique à celui de l'option **-z**.

Constatez la présence du fichier **test.tar.xz** :

```
SUSE12SP1:/ # ls -l /tmp | grep test.tar.xz
-rw-r--r-- 1 root    root    432 Oct 15 08:05 test.tar.xz
```



Important - Notez que le fichier compressé a été créé dans le même répertoire que le fichier source et que le fichier source a disparu. Le fichier source peut être maintenue si l'option **-keep** est spécifiée. Si le fichier test.tar.xz avait déjà existé, la commande aurait échoué avec un message d'erreur. L'extension du fichier est **.xz**, cependant la commande peut aussi gérer l'extension **.lzma**.

Décompressez le fichier test.tar.xz :

```
SUSE12SP1:/ # xz -d /tmp/test.tar.xz
SUSE12SP1:/ # ls -l /tmp | grep test
-rw-r--r-- 1 root    root      512 Oct 15 08:13 test.cpio
-rw-r--r-- 1 root    root    20480 Oct 15 08:05 test.tar
```

Autres Utilitaires

Il existe d'autres utilitaires pour la compression, chacun produisant un fichier ayant une extension spécifique :

Outil	Extension	Commande de Compression	Commande de Décompression
compress	.Z	compress	uncompress
rar	.rar	rar	unrar
zip	.zip	zip	unzip

<html>

Copyright © 2004-2017 I2TCH LIMITED

</html>