La commande dd

Liens web sur la commande dd :

dd et Ubuntu

commande pv : Avoir des informations sur le temps d'une tâche

barre-de-progression-pour-la-commande-dd-

Pour pouvoir copier ou modifier une partition, elle ne doit pas être en cours d'utilisation.

Imaginez que vous cherchiez à copier un livre que l'auteur est en train de modifier où n'a pas encore terminé d'écrire...

Introduction:

Extrait de la page de man de dd :

“dd copie un fichier (par défaut, depuis l'entrée standard vers la sortie standard) en permettant de sélectionner la taille de bloc, et d'effectuer des conversions.”

Utilisée fréquemment, la commande dd est très simple d'utilisation, très puissante et... tout aussi dangereuse. Vérifiez toujours bien les commandes deux (trois ?) fois avant de les lancer.

LES COMMANDES QUI SUIVENT PEUVENT VOUS FAIRE PERDRE TOUTES LES INFORMATIONS CONTENUES DANS VOTRE MACHINE SI ELLES NE SONT PAS CONVENABLEMENT UTILISÉES. VOUS ÊTES PRÉVENUS!

'NOTE:' Pour pouvoir être copiée, déplacée, agrandie ou réduite, la partition source doit au préalable être “démountée” du système (voir “umount”).

Syntaxe

La syntaxe d'une ligne de commandes utilisant dd se présente généralement sous cette forme:

```
    dd if=<source> of=<cible> bs=<taille des blocs> skip= seek=
conv=<conversion>
```

#source = D'où proviennent les données à copier; #cible = où seront copiées les données traitées par la commande; #bs = taille des blocs, de puissance 2, par défaut égale à 512 octets; #skip = Ignorer le nombre indiqué de blocs (dont la taille est fournie par ibs) au début de la lecture; #seek = Ignorer le nombre indiqué de blocs (dont la taille est fournie par ibs) au début de l'écriture. #conv = Modifier le fichier comme indiqué par l'argument conversion, qui peut prendre les valeurs suivantes (pas d'espace autour des virgules lorsque plusieurs arguments sont fournis) :

ascii, ebcdic, ibm, block, unblock, lcase, ucase, swab, noerror, notrunc, sync.

Exemple : Cloner un disque A sur le B

Imaginons, le disque A est la source et le disque B la destination. Il faut que B soit égal ou plus grand que A.

Pour cloner direct de disque à disque, on peut toujours utiliser dd : Copier vers le presse-papierCode BASH :

```
   dd status=progress if=/dev/sda of=/dev/sdb bs=100M
```

Je mets un BS de 100M pour que ça aille plus vite (testé OK sur un SSD, à adapter sur un HDD)

Copier un disque sur un autre avec le materiel Inateck

inateck_copie_dd_001.pdf

Utilisation

Copier une partition d'un disque dur vers un partition d'un autre disque dur

Copie l'intégralité de la partition sdl vers la partition sdm et ne pas interrompre la copie en cas d'erreurs.

```
      sudo dd if=/dev/sdl of=/dev/sdm bs=8192 conv=notrunc,noerror
status=progress
```

256000+0 enregistrements lus 256000+0 enregistrements écrits 1048576000 octets (1,0 GB) copiés, 30,778 s, 34,1 MB/s

noerror = Continuer même en cas d'erreur.

Remarque au sujet de l'UUID.

Copier une partition autre vers une partition copie aussi l'UUID puisqu'il est inscrit dans la partition même.< br /> Vous vous retrouveriez alors avec deux partitions qui ont le même UUID...

```
    blkid
```

/dev/sda9: UUID=“a5355774-2e6b-45fe-8993-0f23602c1616” TYPE=“ext2”

```
   /dev/sdb1: UUID="a5355774-2e6b-45fe-8993-0f23602c1616" TYPE="ext2"
```

Il faut alors modifier l'UUID de la partition copiée pour ne pas avoir de conflit si la partition originale et sa copie sont accessibles au système :

```
# tune2fs -U random /dev/sdb1
# blkid
/dev/sda9: UUID="a5355774-2e6b-45fe-8993-0f23602c1616" TYPE="ext2"
/dev/sdb1: UUID="75aa7a37-d2a2-4cf4-88f8-616067c7a684" TYPE="ext2"
```

Cloner un disque dur en entier

Copie de l'intégralité (y compris le mbr) du disque sdg vers le disque sdh:

```
#dd if=/dev/sdg of=/dev/sdh conv=noerror
```

4030464+0 enregistrements lus 4030464+0 enregistrements écrits 2063597568 octets (2,1 GB) copiés, 1160,41 s, 1,8 MB/s

Copie de l'intégralité (y compris le mbr) du disque sdm vers le disque sdl avec visualisation de la progression:

```
    Autre exemple :
    sudo dd if=/dev/sdm of=/dev/sdl bs=100M status=progress
conv=noerror,sync
```

Réaliser un fichier image ISO copie d'un CD-ROM

```
#dd if=/dev/sr0 of=/home/laurent/cd.iso bs=2048
```

737+0 enregistrements lus 737+0 enregistrements écrits 1509376 octets (1,5 MB) copiés, 3,3215 s, 454 kB/s

Pour monter le fichier image ISO obtenu :

```
# mkdir /mnt/cd
# mount -o loop -t iso9660 cd.iso /mnt/cd
```

Montage permanent par /etc/fstab :

```
/chemin_vers_iso/cd.iso   /mnt/cd  iso9660 rw,user,noauto 0 0
```

Effacer un lecteur

ATTENTION, COMMANDE DANGEREUSE!

Pour les paranos... La commande suivante remplie 7 fois de suite un périphérique avec des données aléatoires. Autant dire qu'il ne sera plus possible de récupérer quoi que ce soit sur ce périphérique...

```
 #for n in $(seq 7); do dd if=/dev/urandom of=/dev/sdc bs=8b; done
```

dd: écriture de « /dev/sdc »: Aucun espace disponible sur le périphérique

```
62465+0 enregistrements lus
62464+0 enregistrements écrits
255852544 octets (256 MB) copiés, 37,0691 s, 6,9 MB/s
dd: écriture de « /dev/sdc »: Aucun espace disponible sur le périphérique
62465+0 enregistrements lus
62464+0 enregistrements écrits
255852544 octets (256 MB) copiés, 37,1506 s, 6,9 MB/s
dd: écriture de « /dev/sdc »: Aucun espace disponible sur le périphérique
62465+0 enregistrements lus
62464+0 enregistrements écrits
255852544 octets (256 MB) copiés, 36,6351 s, 7,0 MB/s
dd: écriture de « /dev/sdc »: Aucun espace disponible sur le périphérique
62465+0 enregistrements lus
62464+0 enregistrements écrits
255852544 octets (256 MB) copiés, 36,4881 s, 7,0 MB/s
dd: écriture de « /dev/sdc »: Aucun espace disponible sur le périphérique
62465+0 enregistrements lus
62464+0 enregistrements écrits
255852544 octets (256 MB) copiés, 36,4908 s, 7,0 MB/s
dd: écriture de « /dev/sdc »: Aucun espace disponible sur le périphérique
62465+0 enregistrements lus
62464+0 enregistrements écrits
255852544 octets (256 MB) copiés, 36,5277 s, 7,0 MB/s
dd: écriture de « /dev/sdc »: Aucun espace disponible sur le périphérique
62465+0 enregistrements lus
62464+0 enregistrements écrits
255852544 octets (256 MB) copiés, 36,897 s, 6,9 MB/s
```

Sauvegarder le chargeur de boot d'un disque dur

```
#dd if=/dev/sda of=ChrgBoot.dd bs=440 count=1
1+0 enregistrements lus
1+0 enregistrements écrits
440 octets (440 B) copiés, 0,000473112 s, 930 kB/s
```

Cette commande copiera les 440 premiers octets du disque dur dans un fichier.

Sauvegarder le MBR d'un disque dur

```
#dd if=/dev/sda of=MBR.dd bs=512 count=1
1+0 enregistrements lus
1+0 enregistrements écrits
512 octets (512 B) copiés, 7,4232e-05 s, 6,9 MB/s
```

Cette commande copiera les 512 premiers octets du disque dur dans un fichier.

Sauvegarder la table des partitions d'un disque dur

```
$ dd if=/dev/sda of=TablePart.dd bs=64 count=1 skip=446
1+0 enregistrements lus
1+0 enregistrements écrits
64 octets (64 B) copiés, 0,000558829 s, 115 kB/s
```

Cette commande copiera les 512 premiers octets du disque dur dans un fichier.

pour restaurer la table de partition

dd if=TablePart.dd of=/dev/sda bs=64 count=1 seek=446

```
64+0 enregistrements lus
64+0 enregistrements écrits
64 octets (64 B) copiés, 0,00132602 s, 48,3 kB/s
```

Cette commande copiera la table des partitions contenue dans le fichier '`TablePart.dd`' sur le disque '`/dev/sda`'

Sauvegarder un disque dur ou une partition dans une image compressée

La ligne de commandes suivante permet de créer un fichier compressé, qui pourra être utilisé pour restaurer la partition “n” (“n” est une valeur numérique) du disque “/dev/sdX” (“X” est une lettre alphabétique).
 (peu importe le type de partition)

```
   #dd if=/dev/sdXn | gzip -c >
/media/suportamovible/fichierimage_part_dd.gz
```

Dans cette ligne de commandes, le programme “dd” lit par blocs (de 512 octets par défaut) le contenu de la partition “n” du disque “X” (/dev/sdXn) Le flux de sortie du programme “dd” est connecté par un tube anonyme (|) au flux d'entrée du programme “gzip”. “gzip” va compresser ce flux et le retourner par son flux de sortie (option “c”) qui sera re-dirigé par “>” vers le fichier “/media/suportamovible/fichierimage_part_dd.gz” Si le fichier “/media/suportamovible/fichierimage_part_dd.gz” existait avant l'exécution de cette ligne de commandes, il serait “écrasé”. Pour restaurer la partition ainsi sauvegardée faire :

```
   #gzip -cd /media/suportamovible/fichierimage_part_dd.gz | dd of=/dev/sdXn
```

Dans cette commande, le programme “gzip” dé-compresse (option “d”) le fichier “/media/suportamovible/fichierimage_part_dd.gz”. Le flux de sortie du programme “gzip” (option “c”) est connecté à un tube anonyme (|) qui est connecté à l'entrée du programme “dd”. La sortie (of=) du programme “dd” se fera sur la partition “n” du disque “X” (/dev/sdXn), et le contenu de cette partition sera écrasé et remplacé par le contenu du fichier sauvegardé par la ligne de commandes précédente.

Clef_usb_d'installation- Graver une ISO sur une clef USB

Commande dd avec barre de defillement

```
 (PV -n image.iso | dd off=/dev/sdb bs=1M && Sync) 2>&1 |dialogue --gauge
"la commande dd est en cours d exécution, merci de patienter..." 10 70 0
```

Références

Page de man dd

Unix DD Command and Image Creation

Commande dd sous Linux

From:
http://chanterie37.fr/fablab37110/ - **Castel'Lab le Fablab MJC de Château-Renault**

Permanent link:
http://chanterie37.fr/fablab37110/doku.php?id=start:parcours_linux:dd

Last update: **2025/07/08 18:09**