|  |  |
| --- | --- |
| Btrfs Filesystem - The Best Thing Since Sliced Bread | Système de fichiers BTRFS |

Table des matières

[Introduction 3](#_Toc81055097)

[Commandes btrfs 4](#_Toc81055098)

[Installation du paquet btrfs 5](#_Toc81055099)

[Création, conversion ou formatage d’une partition avec btrfs 6](#_Toc81055100)

[Sous-volumes 8](#_Toc81055101)

[Création d'un sous-volume 8](#_Toc81055102)

[Montage des sous-volumes 11](#_Toc81055103)

[Autres commandes sur les sous-volumes 12](#_Toc81055104)

[Vérification des erreurs 16](#_Toc81055105)

[Instantannés 18](#_Toc81055106)

[Instantanés d'un sous-volume 19](#_Toc81055107)

[Instantanés en lecture seulement 24](#_Toc81055108)

[Suppression d'un instantané 25](#_Toc81055109)

[Autres fonctionnalités 27](#_Toc81055110)

[Compression 27](#_Toc81055111)

[Options pour les points de montage 31](#_Toc81055112)

[Position des options de montage 31](#_Toc81055113)

[Annexe 01 Utilitaires du système de fichiers btrfs 34](#_Toc81055114)

[Utilitaire df 34](#_Toc81055115)

[Utilitaire btrfs filesystem show|sync 34](#_Toc81055116)

[Utilitaire de défragmentation 35](#_Toc81055117)

[Utilitaire btrfs filesystem resize 35](#_Toc81055118)

[Utilitaire btrfs device 36](#_Toc81055119)

[Utilitaire btrfs scrub 36](#_Toc81055120)

[Utilitaire btrfs filesyetm label 37](#_Toc81055121)

[Annexe 02 Options de montage intéressantes 38](#_Toc81055122)

# Introduction

btrfs (souvent prononcé *butterFS*) est un système de fichiers développé par Oracle et placé sous licence GPL dont le **développement remonte à 2007**.

Au **contraire du système de fichier ext4**, qui est une évolution des systèmes de fichiers ext2/3, btrfs est conçu différemment, et **apporte certaines fonctionnalités inédites**.

Il est livré avec des **fonctionnalités avancées** avec un **accent majeur sur la tolérance aux pannes**, la **redondance** et **l'administration transparente**.

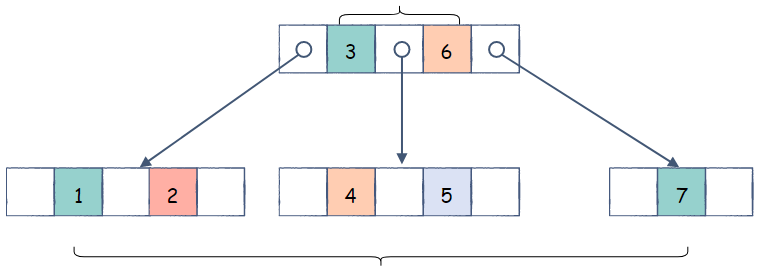
Les **principales fonctionnalités** de btfrs sont les suivantes …

* **Capacité maximale** de **2^64 octets** == **16 Eo maximum** ;
* **Meilleure gestion de l'espace occupé** par les petits fichiers ;
* Possibilité de **créer des instantanés** (*snapshots*) en **lecture seule** ou en **lecture-écriture** ;
* Possibilité de **créer des sous-volumes** ;
* Fonctionnalités de **sommes de contrôle des données et des métadonnées** ;
* Capacité de **compression** (gzip/LZO) ;
* **Sauvegarde incrémentale intégrée** au système de fichiers ;
* Capacité de **défragmentation à chaud** ;
* **Prise en charge des SSD** (stockage flash) (TRIM / Élimination de la notification blocs libres pour réutilisation) et optimisations.

btrfs, une forme abrégée de **B Tree File System**, est un format de stockage informatique qui combine un système de fichiers basé sur le **principe de copie sur écriture** (*Copy-On-Write* ou COW) **avec un gestionnaire de volume logique**.

Il utilise une **structure B-tree** ce qui permet au système de fichiers btrfs de **gérer efficacement de gros morceaux de lecture et d'écriture**.

**Clé**



**Nœuds enfants**

Le système de fichiers btrfs fournit également quelques fonctionnalités intéressantes, telles que les **instantanés** (*snapshots*) et **sous-volumes** (*subvolumes*).  
Un instantané est une **image d'un système de fichiers figé dans le temps**.   
Dans un instantané, le **système de fichiers n’est pas modifié**, même si les **utilisateurs ajoutent et suppriment des fichiers** (le système de fichiers réel est modifié, seul **l’instantané demeure statique**).

L'instantané est utile car il permet aux administrateurs d'effectuer une **sauvegarde du système de fichiers sans** **empêcher les utilisateurs** ou les **processus de travailler sur le système de fichiers**.   
Normalement, une sauvegarde ne peut être effectuée que sur un **système de fichiers statique**, ce qui n’est pas possible pour en hôte en production.

Il a ensuite été **introduit dans le noyau Linux principal en 2009** (noyau version **2.6.29**) et est devenu un système de fichiers **robuste**, **sécurisé** et **fiable**.

Le support btrfs de Debian a été **introduit dans Debian 6** (*Squeeze*).

## Commande btrfs

La commande btrfs **gère et affiche des informations sur le système de fichiers btrfs**.   
Elle prend toujours une sous-commande suivie d'autres arguments de commande.

Sans aucune sous-commande, il affiche son module d’aide ...

**Utilisation  
btrfs [--help] [--version] [--format <format>] [-v|--verbose] [-q|--quiet] <group> [<group>...] <command> [<args>]**

**Groupes de commandes**

|  |  |
| --- | --- |
| Commande | Description |
| balance | Équilibrage des données entre les appareils ou modification des groupes de blocs à l'aide de filtres |
| device | Gestion et interrogation des périphériques dans le système de fichiers |
| filesystem | Affichage des tâches et informations du système de fichiers |
| inspect-internal | Requête diverses informations internes |
| property | Modification des propriétés des objets du système de fichiers |
| qgroup | Modification des propriétés des objets du système de fichiers |
| quota | Gestion des groupes de quotas |
| replace | Remplacement d’un périphérique dans le système de fichiers |
| rescue | Boîte à outils de sauvetage pour des opérations de sauvetage spécifiques |
| scrub | Vérification des sommes de contrôle des données et des métadonnées |
| subvolume | Gestion des sous-volumes (création, suppression, liste, ...) |

**Commandes**

|  |  |
| --- | --- |
| Commande | Description |
| check | Vérification de l'intégrité structurelle d'un système de fichiers (non monté) |
| recieve | Reception des sous-volumes d'un flux |
| restore | Tentative de restauration des fichiers à partir d'un système de fichiers endommagé (non monté) |
| send | Envoie du (des) sous-volume(s) vers la sortie standard (*stdout*) |
| help | Affichage des informations d'aide |
| version | Affichage de la version de btrfs-progs |

Pour un aperçu d'une commande donnée, on utilise btrfs <commande> --help ou btrfs [command...] --help --full pour afficher toutes les options disponibles …  
**>> btrfs subvolume --help**  
**>> btrfs subvolume --help --full**

Tout nom de commande peut être abrégé tant qu'il reste sans ambiguïté.  
Toutefois, il est recommandé d'utiliser des noms de commande complets dans les scripts …  
**>> sudo btrfs subvolume show /mnt/data/projets  
>> sudo btrfs sub sh /mnt/data/projets**

**>> sudo btrfs subvolume list /mnt/data  
>> sudo btrfs sub li /mnt/data  
>> sudo btrfs sub l /mnt/data**

**>> sudo btrfs s l /mnt/data  
btrfs: ambiguous token 's'  
  
Did you mean one of these ?  
 scrub  
 send  
 subvolume**

Tous les groupes de commandes ont leur page de manuel …   
**>> man btrfs-<Groupe>  
>> man btrfs-subvolume**

## Installation du paquet btrfs

Pour la distribution Debian, afin de pouvoir installer le système de fichiers Btrfs sur une partition, on doit **installer le paquet btrfs-progs** …   
**>> sudo apt search btrfs | grep btrfs  
 outil de sauvegarde pour les sous-volumes btrfs  
btrfs-compsize/stable 1.5-1 amd64  
 calcul du ratio de compression d’un ensemble de fichiers dans btrfs  
btrfs-heatmap/stable 9-1 all  
 visualisation de l'organisation des données sur un système de fichiers btrfs  
btrfs-progs/stable,now 5.10.1-2 amd64 [installé]  
…  
>> sudo apt install btrfs-progs**

Pour vérifier l’installation …  
**>> sudo btrfs --version  
btrfs-progs v5.10.1**

ou  
**>> whereis btrfs  
btrfs: /usr/bin/btrfs /usr/share/man/man5/btrfs.5.gz /usr/share/man/man8/btrfs.8.gz**

# Création, conversion ou formatage d’une partition avec btrfs

Avant d’installer le système de fichier (formater) un disque à l’aide de Btrfs, on doit le partitionner.  
Pour ce faire, on utilise les **utilitaires fdisk ou parted**.

Pour installer le système d’exploitation sur une partition avec le système de fichier btrfs …   
**>> sudo mkfs.btrfs <Partition >**  
**>> sudo** **mkfs.bttrfs /dev/sdb1**   
où /dev/sdb1 est la partition que l’on souhaite formater.   
**btrfs-progs v4.7.3**   
**See http://btrfs.wiki.kernel.org for more information.**   
  
**Label:** **(null)**   
**UUID:**   
**Node size:** **16384**   
**Sector size:** **4096**   
**Filesystem size:** **20.00GiB**   
**Block group profiles:**   
 **Data:** **single** **8.00MiB**   
 **Metadata:** **DUP** **1.00GiB**   
 **System:** **DUP** **8.00MiB**   
**SSD detected:** **no**   
**Incompat features:** **extref, skinny-metadata**   
**Number of devices:** **1**   
**Devices:**   
 **ID** **SIZE PATH**   
 **20.00GiB** **/dev/sdb**

Pour **convertir une partition existante** avec un **système de fichier déjà installé** …   
**>> sudo btrfs-convert <Partition >  
>> sudo** **btrfs-convert /dev/sdb1**   
**create btrfs filesystem:  
 blocksize: 4096  
 nodesize: 16384  
 features: extref, skinny-metadata (default)  
 checksum: crc32c  
free space report:  
 total: 19998441472  
 free: 19187040256 (95.94%)  
creating ext2 image file  
creating btrfs metadata  
copy inodes [o] [ 2/ 11]  
conversion complete**

Comme dernière étape, il faut **ajouter un point de montage au fichier /etc/fstab** …   
**/dev/sdb1 /mnt/data btrfs defaults 0 0**

Pour relire le fichier /etc/fstab …  
**>> sudo mount --all**

Il est également possible de créer un **point de montage systemd**.

Pour créer le fichier du point de montage …  
**>> sudo mkdir /mnt/data**

**>> sudo nano /etc/systemd/system/mnt-data.mount**

**[Unit]**

**Description = Partage btrfs**

**[Mount]**

**What = /dev/sdb1**

**Where = /mnt/data**

**Type = btrfs**

**[Install]**

**WantedBy = multi-user.target**

**Attention** …  
Le nom du fichier de configuration systemd (**/etc/systemd/system/mnt-data.mount**) est en fonction du répertoire local (**/mnt/data**).

Ce nom de fichier systemd doit **obligatoirement** refléter le nom du répertoire.

Il faut par la suite actualiser le démon systemd …

**>> sudo systemctl daemon-reload**

Comme dernière étape, il faut démarrer le point de montage …  
**>> sudo systemctl start mnt-data.mount**

Pour que le point de montage soit activé à chaque démarrage de la station …

**>> sudo systemctl enable mnt-data.mount**

**Created symlink /etc/systemd/system/multi-user.target.wants/mnt-data.mount → /etc/systemd/system/mnt-data.mount.**

Pour vérifier le point de montage …  
Il faut ensuite renseigner le système d’un nouveau point de montage …   
**>> sudo mount  
…  
/dev/sdb1 on /mnt/data type btrfs (rw,relatime,space\_cache,subvolid=5,subvol=/)**

# Sous-volumes

Un **sous-volume btrfs n'est pas un périphérique de bloc** (et ne peut pas être traité comme un disque).  
À la place, un sous-volume btrfs peut être considéré comme **un espace de noms (*namespsace*) de fichier POSIX**.   
Cet espace de noms est **accessible à l’aide du sous-volume de niveau supérieur** du système de fichiers, ou il peut être **monté de son propre chef**.

Chaque système de fichiers btrfs possède un **sous-volume de niveau supérieur avec l'ID 5**.   
Il peut être **monté en tant que /** (par défaut), ou un autre sous-volume peut être monté à la place.

Les sous-volumes **peuvent être déplacés dans le système de fichiers** et sont plutôt **utilisés par leur identifiant** **que par leur chemin**.

Les **sous-volumes peuvent être montés comme des partitions de système de fichiers** à l'aide des indicateurs de montage.   
Par exemple, on pourrait avoir un sous-volume nommé et le monter en tant que tel.

On peut **imiter les partitions traditionnelles du système de fichiers** en créant divers sous-volumes sous le niveau supérieur du système de fichiers, puis en les montant aux points de montage appropriés.

Ainsi, on peut facilement restaurer un système de fichiers (ou une partie de celui-ci) à un état antérieur en utilisant des instantanés (*snapshots*).

## Création d'un sous-volume

Conditions préalables pour la suite de ce tutoriel …

* On utilise la partition /dev/sdb1 ;
* On a installé le système de fichier btrfs sur cette partition ;
* On a monté /dev/sdb1 dans le répertoire /mtn/data.

Pour créer un sous-volume …  
**>> sudo btrfs subvolume create <Chemin du sous-volume>**

**>> sudo btrfs subvolume create /mnt/data/photos  
>> sudo btrfs subvolume create /mnt/data/videos  
>> sudo btrfs subvolume create /mnt/data/documents  
>> sudo btrfs subvolume create /mnt/data/projets**

Pour vérifier l’arborescence …  
**>> sudo tree /mnt/data/  
/mnt/data/  
 documents  
 photos  
 projets  
 videos**

**Remarque** …  
Il est possible que l’on doive installer l’utilitaire tree …  
**>> sudo apt install tree**

Il est possible de lister tous les sous-volumes du système de fichiers (monté sur le répertoire /mnt/data) …  
**>> sudo btrfs subvolume list /mnt/data  
ID 257 gen 17 top level 5 path photos  
ID 258 gen 17 top level 5 path videos  
ID 259 gen 17 top level 5 path documents  
ID 260 gen 17 top level 5 path projets**

Comme on peut le constater, tous les sous-volumes qui ont été créés.

On peut **afficher de nombreuses informations sur un sous-volume** (comme /mnt/data/projects ) comme …

* le **nom du sous-volume** ;
* l'**UUID** **du sous-volume** ;
* l'**ID** **du sous-volume** ;
* **…**

**>> sudo btrfs subvolume show /mnt/data/projets  
projets  
 Name: projets  
 UUID: da630784-6c2e-f84c-9362-a9485595e642  
 Parent UUID: -  
 Received UUID: -  
 Creation time: 2021-08-24 14:51:44 -0400  
 Subvolume ID: 260  
 Generation: 17  
 Gen at creation: 16  
 Parent ID: 5  
 Top level ID: 5  
 Flags: -  
 Snapshot(s):**

On crée quelques fichiers dans chacun des sous-volumes btrfs.   
Une fois que l’on a monté les sous-volumes btrfs séparément, les fichiers de chacun des sous-volumes devraient être là.

Pour créer quelques fichiers dans le sous-volume /mnt/data/projets …  
**>> sudo touch /mnt/data/projets/fichier0{1..3}  
>> sudo touch /mnt/data/documents/fichier0{4..5}  
>> sudo touch /mnt/data/videos/fichier0{6..7}   
>> sudo touch /mnt/data/photos/fichier0{8..9}  
>> sudo tree /mnt/data**

**/mnt/data**

**documents**

**fichier04**

**fichier05**

**photos**

**fichier08**

**fichier09**

**projets**

**fichier01**

**fichier02**

**fichier03**

**videos**

**fichier06**

**fichier07**

On peut **monter un sous-volume en utilisant son ID**.  
Auparavant on doit créer le répertoire /tmp/projets …   
**>> sudo mkdir --verbose /tmp/projets**

On peut ensuite monter le sous-volume projets (disponible sur la partition sdb1) en utilisant son ID 260 dans le répertoire /tmp/projets …  
**>> sudo mount /dev/sdb1 --options subvolid=260 /tmp/projets**ou **>> sudo mount /dev/sdb1 -o subvolid=260 /tmp/projets**

Le sous-volume est monté dans le répertoire /tmp/projets …  
**>> sudo btrfs subvolume show /tmp/projets  
projets  
 Name: projets  
 UUID: da630784-6c2e-f84c-9362-a9485595e642  
 Parent UUID: -  
 Received UUID: -  
 Creation time: 2021-08-24 14:51:44 -0400  
 Subvolume ID: 260  
 Generation: 21  
 Gen at creation: 16  
 Parent ID: 5  
 Top level ID: 5  
 Flags: -  
 Snapshot(s):**

Il est également de voir que le système de fichiers (sous-volume projets) est monté sur le répertoire /tmp/projets ...  
**>> sudo df --human-readable --type=btrfs**ou  
**>> sudo df -h -t btrfs  
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur  
/dev/sdb1 19G 3,5M 19G 1% /mnt/data  
/dev/sdb1 19G 3,5M 19G % /tmp/projets**

Tous les fichiers du sous-volume projets sont également disponibles dans le répertoire /tmp/projets …  
**>> sudo tree /tmp/projets  
/tmp/projets  
 fichier01  
 fichier02  
 fichier03**

**Astuce** …  
La modification de la **disposition des sous-volumes est simplifiée** en **n'utilisant pas le sous-volume de niveau supérieur** (ID=5) comme / (ce qui est fait par défaut).   
Au lieu de cela, il est intéressant de **créer un sous-volume afin d’héberger ses données réelles** et de le **monter en tant** **que fichier** **/**.

## Montage des sous-volumes

Pour **monter un sous-volume**, on doit connaître soit **son nom**, soit **son ID** …  
**>> sudo btrfs subvolume list /mnt/data  
ID 257 gen 23 top level 5 path photos  
ID 258 gen 23 top level 5 path videos  
ID 259 gen 23 top level 5 path documents  
ID 260 gen 21 top level 5 path projets**

On peut alors monter le sous-volume /mnt/data/projets.   
Il possède l'ID 260.  
Il sera monté dans le répertoire /tmp/projets.

On crée un répertoire /tmp/projects …  
**>> sudo mkdir --verbose /tmp/photos  
mkdir: création du répertoire '/tmp/photos'**

On peut monter le sous-volume /mnt/data/projets (disponible sur la partition sdb1) en utilisant son nom dans le répertoire /tmp/projets …  
**>> sudo** **mount /dev/sdb1 -o subvol=photos /tmp/photos**

Le sous-volume projets est monté dans le répertoire /tmp/projects …  
**>> sudo** **btrfs subvolume show /tmp/photos  
projets  
 Name: photos  
 UUID: da630784-6c2e-f84c-9362-a9485595e642  
 Parent UUID: -  
 Received UUID: -  
 Creation time: 2021-08-24 14:51:44 -0400  
 Subvolume ID: 260  
 Generation: 21  
 Gen at creation: 16  
 Parent ID: 5  
 Top level ID: 5  
 Flags: -  
 Snapshot(s):**

Il est également possible de voir que le système de fichiers (sous-volume projets) est monté dans le répertoire /tmp/projets ...  
**>> sudo** **df --human-readable --type=btrfs**ou   
**>> sudo** **df -h -t btrfs  
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur  
/dev/sdb1 19G 3,5M 19G 1% /mnt/data  
/dev/sdb1 19G 3,5M 19G % /tmp/projets  
/dev/sdb1 19G 3,5M 19G % /tmp/photos**

Tous les fichiers que créés dans le sous-volume projets sont également disponibles dans le répertoire /tmp/projets …  
**>> sudo tree** **/tmp/photos  
/tmp/projets  
 fichier08  
 fichier09**

**Remarque** …  
La **plupart des options de montage s'appliquent à l'ensemble du système de fichiers** et seules les **options du premier sous-volume à monter prendront effet**.   
Cela est dû à un manque de mise en œuvre et cela peut évoluer avec le temps.

## Autres commandes sur les sous-volumes

### Liste des sous-volumes

Pour **afficher une liste des sous-volumes actuels** et **leurs identifiants** …  
**>> sudo** **btrfs subvolume list <Répertoire>   
>> sudo** **btrfs subvolume list /mnt/data  
ID 257 gen 23 top level 5 path photos  
ID 258 gen 23 top level 5 path videos  
ID 259 gen 23 top level 5 path documents  
ID 260 gen 21 top level 5 path projets**

**>> sudo** **btrfs subvolume list -p <Répertoire>   
>> sudo** **btrfs subvolume list -p /mnt/data  
ID 257 gen 23 parent 5 top level 5 path photos  
ID 258 gen 23 parent 5 top level 5 path videos  
ID 259 gen 23 parent 5 top level 5 path documents  
ID 260 gen 21 parent 5 top level 5 path projets**

### Suppression d’un sous-volume

Pour supprimer un sous-volume …  
**>> sudo** **btrfs subvolume delete <Sous-volume>   
Remarque** …  
Depuis Linux 4.18, on peut également supprimer un sous-volume comme un répertoire ordinaire (rm -r ou rmdir) …  
**>> sudo** **rm -r <Répertoire>**

On crée un sous- volume /mnt/data/test …  
**>> sudo** **btrfs subvolume create /mnt/data/test  
Create subvolume '/mnt/data/test'  
>> sudo** **btrfs subvolume list /mnt/data/test  
ID 257 gen 23 top level 5 path photos  
ID 258 gen 23 top level 5 path videos  
ID 259 gen 23 top level 5 path documents  
ID 260 gen 21 top level 5 path projets  
ID 262 gen 24 top level 5 path test**

Pour supprimer le sous-volume test …  
**>> sudo** **btrfs subvolume delete /mnt/data/test  
Delete subvolume (no-commit): '/mnt/data/test'**

**Remarque** …  
Si on **supprime un sous-volume**, tous les **fichiers/répertoires de ce sous-volume seront également supprimés**.

### Montage de sous-volumes au démarrage

Afin de monter les sous-volumes du système de fichiers Btrfs, on utilise le fichier /etc/fstab ou un point de montage systemd.

Il faut, dans un premier temps, savoir si le point de montage est actif …  
**>> sudo** **mount  
…  
/dev/sdb1 on /mnt/data type btrfs (rw,relatime,space\_cache,subvolid=5,subvol=/)  
/dev/sdb1 on /tmp/projets type btrfs (rw,relatime,space\_cache,subvolid=260,subvol=/projets)  
/dev/sdb1 on /tmp/photos type btrfs (rw,relatime,space\_cache,subvolid=257,subvol=/photos)**

On démonte le système de fichiers Btrfs actif sur le répertoire /mnt/data …  
**>> sudo umount /mnt/data  
>> sudo** **mount  
…  
/dev/sdb1 on /tmp/photos type btrfs (rw,relatime,space\_cache,subvolid=257,subvol=/photos)   
/dev/sdb1 on /tmp/projets type btrfs (rw,relatime,space\_cache,subvolid=260,subvol=/projets)**

On doit créer les répertoires qui serviront de point de montage …  
**>> sudo** **mkdir -pv /mnt/data/{documents,projets,photos,videos}  
mkdir: création du répertoire '/mnt/data/documents'  
mkdir: création du répertoire '/mnt/data/projets'  
mkdir: création du répertoire '/mnt/data/photos'  
mkdir: création du répertoire '/mnt/data/videos'**

On **recherche l'UUID du système de fichiers Btrfs** sur la partition sdb1 …  
**>> sudo blkid /dev/sdb1  
/dev/sdb1: UUID="17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e" UUID\_SUB="dfe601e7-6d66-4bff-b115-f69f0ba4fd76" BLOCK\_SIZE="4096" TYPE="btrfs" PARTUUID="fd7c207d-60a8-4c1d-bcec-585ad8a57dea"**

**Remarque** …  
Il est important de faire une **copie du fichier original de /etc/fstab** avant de le modifier …  
**>> sudo cp -a /etc/fstab /etc/fstab.original**

On édite le fichier /etc/fstab …  
**>> sudo** **nano /etc/fstab**

On ajoute les lignes suivantes sous la forme …  
**<UUID> <Point de montage> btrfs subvol=<Sous-volume> 0 0**

**# Montage les sous-volumes Btrfs dans leurs répertoires respectifs  
UUID="1d1ade56-fbe6-4ce1-b066-6881d11f264e" /mnt/data btrfs defaults 0 0  
UUID="1d1ade56-fbe6-4ce1-b066-6881d11f264e" /mnt/data/projets btrfs subvol=projets 0 0  
UUID="1d1ade56-fbe6-4ce1-b066-6881d11f264e" /mnt/data/documents btrfs subvol=documents 0 0  
UUID="1d1ade56-fbe6-4ce1-b066-6881d11f264e" /mnt/data/photos btrfs subvol=photos 0 0  
UUID="1d1ade56-fbe6-4ce1-b066-6881d11f264e" /mnt/data/videos btrfs subvol=videos 0 0**

**Remarque** …  
On apporte les **modifications nécessaires en fonction de son UUID**.

On doit relire le fichier /etc/fstab afin que le système prenne en compte les nouveaux points de montage …  
**>> sudo** **mount --all**ou **>> sudo** **mount -a**

On peut également vérifier si les points de montage sont actifs à l’aide de la commande df …  
**>> df --human-readable --type=btrfs**ou **>> df -h -t btrfs  
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur  
/dev/sdc1 19G 3,5M 19G 1% /mnt/data/photos  
/dev/sdc1 19G 3,5M 19G 1% /mnt/data  
/dev/sdc1 19G 3,5M 19G 1% /mnt/data/videos  
/dev/sdc1 19G 3,5M 19G 1% /mnt/data/documents  
/dev/sdc1 19G 3,5M 19G 1% /mnt/data/projets**

Il est également possible de monter les sous-volumes du système de fichiers Btrfs à l’aide d’un **point de montage systemd** …  
**>> sudo nano /etc/systemd/system/mnt-data-projets.mount**

**[Unit]  
Description=Point de montage pour Projets**

**[Mount]  
What=/dev/disk/by-uuid/17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e  
Where=/mnt/data/projets  
Type=btrfs  
Options=subvol=projets**

**[Install]  
WantedBy=multi-user.target**

**Attention** …  
Le nom du fichier de configuration systemd (**/etc/systemd/system/mnt-data-projets.mount**) est en fonction du répertoire local (**/mnt/data/projets**).

Ce nom de fichier systemd doit **obligatoirement** refléter le nom du répertoire.

Il faut par la suite actualiser le démon systemd …

**>> sudo systemctl daemon-reload**

Comme dernière étape, il faut démarrer le point de montage …  
**>> sudo systemctl start mnt-data-projets.mount**

Pour que le point de montage soit activé à chaque démarrage de la station …

**>> sudo systemctl enable mnt-data-projets.mount**

**Created symlink /etc/systemd/system/multi-user.target.wants/mnt-data.mount → /etc/systemd/system/mnt-data.mount.**

Pour vérifier le point de montage …  
Il faut ensuite renseigner le système d’un nouveau point de montage …   
**>> sudo mount  
…  
/dev/sdc1 on /mnt/data/projets type btrfs (rw,relatime,space\_cache,subvolid=260,subvol=/projets)**

**Remarque** …  
Il faut modifier le fichier /etc/fstab en conséquence afin de ne pas monter deux fois le même sous-volume.

### Vérification de l'utilisation de l'espace disque

Avec le système de fichiers btrfs, vérifier l'utilisation de l'espace disque à l'aide de la **commande df peut être trompeur** lorsqu'un **système de fichiers est monté et que des fichiers y sont copiés**.

Pour **obtenir des informations ou des résultats plus précises**, on utilise la **commande btrfs fileststem df** …  
**>> sudo btrfs filesystem df <point de montage>  
>> sudo btrfs filesystem df /mnt/data/projets  
Data, single: total=32.00MiB, used=0.00B  
System, single: total=4.00MiB, used=16.00KiB  
Metadata, single: total=32.00MiB, used=176.00KiB  
GlobalReserve, single: total=3.25MiB, used=0.00B**

### Affichage de la structure du système de fichiers

Le commutateur show permet de sonder la structure de fichiers du système de fichiers ou du sous-volume.

Pour afficher la structure de fichiers du système de fichiers …  
**>> sudo btrfs filesystem show <point de montage>  
>> sudo btrfs filesystem show /mnt/data/projets  
Label: none uuid: 17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e  
 Total devices 1 FS bytes used 192.00KiB  
 devid 1 size 18.62GiB used 68.00MiB path /dev/sdb1**

### Synchronisation du système de fichiers

La synchronisation du système de fichiers btrfs permet de **vider les données de la mémoire sur le disque** - ce qui une fois terminé - **conduirait à des données cohérentes** **plus récentes sur le disque**.

Pour forcer la synchronisation du système de fichiers, on **utilise le commutateur de synchronisation**.   
Le **système de fichiers doit déjà être monté** pour que le processus de synchronisation se déroule avec succès.

Pour synchroniser le système de fichiers …  
**>> sudo btrfs filesystem sync <point de montage>  
>> sudo btrfs filesystem sync /mnt/data/projets**

## Vérification des erreurs

Puisque btrfs stocke les **sommes de contrôle** (*checksum*), il est donc toujours capable de **vérifier si** **un fichier**   
(ou des métadonnées) **est en bon état ou s'il a été corrompu**.

La plupart des autres systèmes de fichiers comme **ext4 n'ont pas de somme de contrôle**, ils **n’empêcheront donc pas de lire un fichier qui a été corrompu** par un mauvais périphérique (qui est sur le point de mourir et qui a déjà commencé à corrompre les données).   
Il s'agit d'une fonctionnalité importante de protection des données dans btrfs et cela rend le nettoyage possible.

L’utilisation de la **commande btrfs scrub effectue un nettoyage des données** de la partition au complet. Fondamentalement, elle **lit toutes les données sur le disque**, **recalcule sa somme de contrôle** et **compare la somme de** **contrôle recalculée à celle stockée**.   
Lorsque les **sommes de contrôle stockées** **et recalculées ne correspondent pas**, le système sait qu'il y a **corruption**.

La raison pour laquelle ce nettoyage est important est que les **disques durs ne sont fiables qu'à +/- 99,9999999999999%**.   
Ainsi, sur le nombre de téraoctets d'Entrée/Sortie de données, il est probable qu'il se produise des erreurs.

Bien que ces **erreurs puissent être et soient détectées** (et **corrigées, en supposant qu'une copie redondante** est toujours valide) lors d'un accès disque normal, le **nettoyage de routine du disque complet** est capable de **trouver et de** **corriger les erreurs** **avant que toutes les copies des mêmes données ne soient corrompues**.

La **commande btrfs scrub** **lit toutes les données**, **pas uniquement les fichiers**.   
Cela inclut les **métadonnées**, mais aussi les **copies secondaires de fichiers** qui ne seraient normalement pas dans le chemin de lecture.   
Lorsqu'il lit ces copies secondaires, cela crée une **opportunité pour la correction d'erreurs**, de **démarrer** et de **restaurer** **les données** à partir d'une **copie redondante**.

Pour **effectuer une vérification** du système de fichiers …  
**>> sudo btrfs scrub start <point de montage>  
>> sudo btrfs scrub start /mnt/data/projets  
scrub started on /mnt/data/projets, fsid 17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e (pid=14857)**

**>> sudo btrfs scrub status <point de montage>  
>> sudo btrfs scrub status /mnt/data/projets  
UUID: 17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e  
Scrub started: Wed Aug 25 12:38:02 2021  
Status: finished  
Duration: 0:00:00  
Total to scrub: 192.00KiB  
Rate: 0.00B/s  
Error summary: no errors found**

Pour **annuler l'exécution d’une vérification** scrub …  
**>> sudo btrfs scrub cancel <point de montage>  
>> sudo btrfs scrub cancel /mnt/data/projets  
ERROR: scrub cancel failed on /mnt/data/projets: not running**

Si la commande btrfs scrub s'est **exécuté avec succès sans aucun problème et s'est terminé**, le système affichera un **message d’erreur** indiquant que l'**annulation** **a échouée** car la vérification n'est pas en cours d'exécution.

Pour **reprendre ou poursuivre une vérification interrompue** …  
**>> sudo btrfs scrub resume <point de montage>  
>> sudo btrfs scrub resume /mnt/data/projets**

Encore une fois, si la commande de nettoyage est exécutée avec succès sans interruption, le système indiquera qu'il n'y a rien à reprendre.

# Instantannés

Le système de fichiers btrfs propose une **prise en charge intégrée des instantanés** au niveau du système de fichiers.

Il est possible de **créer un sous-volume** pour un système de fichiers btrfs et **prendre des instantanés des** **fichiers/répertoires** de ce sous-volume.   
Prendre un instantané d'un sous-volume **enregistrera l'état des fichiers/répertoires** dans ce sous-volume.

On pourra **éventuellement récupérer tous les fichiers/répertoires** du sous-volume **à partir de l'instantané** au cas où on en aurait besoin.

La fonction d'instantané du système de fichiers Btrfs utilise le **principe *Copy-on-Write* (CoW)**.   
Ainsi, un instantané **ne prend pas beaucoup d'espace disque** et il est possible de prendre des instantanés d'un sous-volume en un instant.

Le système de fichiers Btrfs prend en charge **2 types d'instantanés** …

* **Instantanés inscriptibles**À l’aide d’un instantané inscriptible, on peut **modifier les fichiers/répertoires** de cet instantané.   
  Il s'agit du **type d'instantané par défaut** du système de fichiers btrfs ;
* **Instantanés en lecture seule**À l’aide d’un instantané en lecture seule, **on ne peut pas modifier les fichiers/répertoires** de cet instantané.

Un instantané est donc une **image du système de fichiers** ou d'un de ses **sous-volumes que l’on gèle à un instant précis**, dans le **but de pouvoir y accéder ou la restaurer plus tard**.

Une fois créé, un **instantané se comporte comme un répertoire ordinaire** de son système de fichiers …  
on **peut le copier**, **le déplacer** vers un support externe, **le renommer** …

**Remarque** …  
Avec le système de fichiers btrfs, il est **uniquement possible de prendre des instantanés des sous-volumes**.   
La **racine principale** d'un système de fichiers btrfs **est également un sous-volume**.   
il est ainsi possible d’effectuer la sauvegarde de l'ensemble du système de fichiers ou de sous-volumes spécifiques.

**Rappel de la mise en place**

Pour créer un système de fichiers Btrfs avec une étiquette sur la partition sdb1 …  
**>> sudo mkfs.btrfs -L data /dev/sdb1**

Pour créer un répertoire /mnt/data …  
**>> sudo mkdir --volume /mnt/data**

Pour monter le système de fichiers Btrfs créé sur la partition sdb1 dans le répertoire /mnt/data …  
**>> sudo mount /dev/sdb1 /mnt/data**

Pour vérifier si le système de fichier est monté …  
**>> df --human-readable /mnt/data**ou **>> df -h /mnt/data**

## Instantanés d'un sous-volume

À la suite de la création d’un instantané …

* les **modifications faites dans chacun des deux répertoires sont indépendantes** ;
* Le système de fichiers gère les modifications pour qu'elles soient **transparentes aux yeux de l'utilisateur** (avec ext4, il aurait fallu créer une copie du répertoire, ce qui aurait occupé le double d'espace disque ;
* Un instantané ne **consomme pas d'espace disque supplémentaire**.

On crée un nouveau répertoire /mnt/data/documents, si cela n’est déjà fait …  
**>> sudo mkdir -v /mnt/data/documents**

On met en place un nouveau sous-volume web1 dans le répertoire /mnt/data/documents …  
**>> sudo btrfs subvolume create /mnt/data/documents/web1**

On ajoute un nouveau fichier index.html dans le sous- volume /mnt/data/documents/web1 comme suit :  
**>> sudo nano /mnt/data/documents/web1/index.html**

On entre les lignes de codes suivantes dans le fichier index.html …  
**<!DOCTYPE html>  
<html>  
 <head>  
 <!-- En-tête de la page -->  
 <meta charset="utf-8" />  
 <title>En construction</title>  
 </head>  
  
 <body>  
 <!-- Corps de la page -->  
 Site en construction  
 </body>  
</html>**

On ajoute un nouveau fichier style.css dans le sous-volume /mnt/data/documents/web1 …  
**>> sudo nano /mnt/data/documents/web1/style.css  
h1 {  
color: green;  
}**

Le sous-volume /mnt/data/documents/web1 contient les fichiers index.html et style.css …  
**>> sudo ls -lh /mnt/data/documents/web1**

On **souhaite conserver tous les instantanés** de ce système de fichiers Btrfs dans le **répertoire /mnt/data/.snapshots**.  
On crée le répertoire /mnt/data/.snapshots …  
**>> sudo mkdir -v /mnt/data/.snapshots  
mkdir: création du répertoire '/mnt/data/.snapshots'**

Pour **prendre un instantané du sous-volume** /mnt/data/documents/web1   
dans le répertoire /mnt/data/.snapshots/web1-version01 (sera créé automatiquement) …  
**>> sudo btrfs subvolume snapshot /mnt/data/documents/web1 /mnt/data/.snapshots/web1-version01**

Un instantané du répertoire /mnt/data/projets/web1 est créé dans le répertoire /mnt/data/.snapshots/web1-version01.

Comme on peut le constater un **nouveau sous-volume .snapshots/web1-version01 a été créé**.   
Un **instantané est en fait un sous-volume** …  
**>> sudo btrfs subvolume list /mnt/data  
…  
ID 263 gen 20 top level 5 path .snapshots/web1-version01**

On peut voir **plus d'informations sur l'instantané créé** dans le répertoire /mnt/data/.snapshots/web1-version01 …  
**>> sudo btrfs subvolume show /mnt/data/.snapshots/web1-version01  
.snapshots/web1-version01  
 Name: web1-version01  
 UUID: 073d578f-21a4-8a4b-b60a-2b01e042db81  
 Parent UUID: a3eea43b-3c7f-4d4b-b65f-f585b5760eb5  
 Received UUID: -  
 Creation time: 2021-08-27 10:14:46 -0400  
 Subvolume ID: 263  
 Generation: 20  
 Gen at creation: 20  
 Parent ID: 5  
 Top level ID: 5  
 Flags: -  
 Snapshot(s):**

Tous les fichiers qui se trouvent dans le sous-volume /mnt/data/documents/web1 se retrouvent également au niveau de l'instantané /mnt/data/.snapshots/web1-version01.  
**>> tree -a /mnt/data/  
/mnt/data/**

**documents**

**fichier04**

**fichier05  
 web1**

**index.html**

**style.css**

**.snapshots**

**web1-version01**

**index.html**

**style.css**

**photos**

**fichier08**

**fichier09**

**projets**

**fichier01**

**fichier02**

**fichier03**

**videos**

**fichier06**

**fichier07**

### Récupération d’un fichier à d’un instantané.

Les instantanés apparaissent comme des répertoires que l’on peut manipuler à sa guise.

On ouvre le fichier /mnt/data/documents/web1/index.html avec l'éditeur de texte nano …  
**>> sudo nano /mnt/data/documents/web1/index.html**

On apporte des modifications au fichier.  
**<!DOCTYPE html>  
<html>  
 <head>  
 <!-- En-tête de la page -->  
 <meta charset="utf-8" />  
 <title>Site en construction</title>  
 </head>  
  
 <body>  
 <!-- Corps de la page -->  
 Site en construction -- Merci de revenir bientôt  
 </body>  
</html>**

Par la suite, on réalise que les modifications apportées au fichier index.html principal sont indésirables et on souhaite **récupérer le fichier index.html à partir de l'instantané** …  
**>> ls -l /mnt/data/documents/web1/index.html  
-rw-r--r-- 1 root root 277 27 aoû 10:23 /mnt/data/projets/web1/index.html  
>> ls -l /mnt/data/.snapshots/web1-version01/index.html  
-rw-r--r-- 1 root root 243 27 aoû 10:08 /mnt/data/.snapshots/web1-version01/index.html**

On **copie le fichier de l’instantané** dans le répertoire original …  
**>> sudo cp -v /mnt/data/.snapshots/web1-version01/index.html /mnt/data/documents/web1/index.html  
'/mnt/data/.snapshots/web1-version01/index.html' -> '/mnt/data/projets/web1/index.html'**

**Remarque** …  
On utilise simplement la commande mv.

Comme on peut le constater, le **fichier index.html est restauré à partir de l'instantané** …  
**>> ls -l /mnt/data/documents/web1/index.html  
-rw-r--r-- 1 root root 243 27 aoû 10:28 /mnt/data/projets/web1/index.html**

### Récupération de données supprimées

De la même manière, il est possible de récupérer, à partir d’un instantané, de récupérer des fichiers qui ont été supprimé.

On supprime tous les fichiers du répertoire /mnt/data/documents/web1 …  
**>> sudo** **rm --recursive --verbose /mnt/data/documents/web1/\*  
'/mnt/data/documents/web1/index.html' supprimé  
'/mnt/data/documents/web1/style.css' supprimé**

**Remarque** …  
Pour l’étape qui suit, il est possible que l’utilitaire rsync ne soit pas installé …  
**>> sudo apt install** **rsync**

Pour **récupérer tous les fichiers/répertoires** depuis l'instantané …  
**>> sudo** **rsync --archive --verbose --compress --delete /mnt/data/.snapshots/web1-version01/ /mnt/data/documents/web1/**ou **>> sudo** **rsync -avz --delete /mnt/data/.snapshots/web1-version01/ /mnt/data/documents/web1/  
/data/documents/web1/  
sending incremental file list  
./  
index.html  
style.css  
sent 376 bytes received 57 bytes 866.00 bytes/sec  
total size is 264 speedup is 0.61**

**Remarque** …  
Les détails de la commande rsync sont …

* **--archive**  
  Mode Archivage ;
* **--verbose**  
  Augmentation de la verbosité ;  
  **--compress**  
  Compression des données du fichier pendant le transfert ;
* **--delete**  
  Suppression des fichiers existants dans le répertoire de destination

Tous les fichiers et répertoires du sous-volume /mnt/data/documents/web1 ont été restaurés (en mode miroir) à partir de l’instantané /mnt/data/.snapshots/web1-version01 .

L'arborescence des fichiers du sous-volume /mnt/data/documents/web1 et de l'instantané /mnt/data/.snapshots/web1-version01 doit être la même …  
**>> sudo tree -a /mnt/data**

**/mnt/data**

**…**

**documents**

**fichier01**

**fichier02**

**fichier03**

**web1**

**index.html**

**style.css**

**.snapshots**

**web1-version01**

**index.html**

**style.css**

**…**

### Mise à jour d'un instantané

Par défaut, le système de fichiers Btrfs prend des instantanés inscriptibles.   
Un instantané Btrfs est comme un sous-volume.   
Ainsi, il est possible de modifier/mettre à jour les fichiers/répertoires d'un instantané inscriptible.

­On modifie le fichier index.html dans le sous- volume /mnt/data/documents/web1 …  
**>> sudo nano /mnt/data/documents/web1/index.html  
<!DOCTYPE html>  
<html>  
 <head>  
 <!-- En-tête de la page -->  
 <meta charset="utf-8" />  
 <title>Site en construction</title>  
 </head>  
  
 <body>  
 <!-- Corps de la page -->  
 Site en construction – Disponible début septembre  
 </body>  
</html>**

On souhaite conserver le fichier index.html du sous-volume /mnt/data/documents/web1.

Pour mettre à jour le fichier index.html de l'instantané /mnt/data/.snapshots/web1-version01 …  
**>> sudo cp -v /mnt/data/documents/web1/index.html /mnt/data/.snapshots/web1-version01/index.html**

On peut vérifier le bon fonctionnement …  
**>> sudo cat /mnt/data/documents/web1/index.html  
<!DOCTYPE html>  
<html>  
 <head>  
 <!-- En-tête de la page -->  
 <meta charset="utf-8" />  
 <title>Site en construction</title>  
 </head>  
  
 <body>  
 <!-- Corps de la page -->  
 Site en construction -- Disponible début septembre  
 </body>  
</html>**

**>> sudo cat /mnt/data/.snapshots/web1-version01/index.html  
<!DOCTYPE html>  
…  
 <title>Site en construction</title>  
…  
 Site en construction -- Disponible début septembre  
…**

La mise à jour d'un instantané est **aussi simple que de copier de nouveaux fichiers sur l'instantané**.

## Instantanés en lecture seulement

Parfois, on ne veut pas que les instantanés puissent être mis à jour de quelque manière que ce soit.   
Dans ce cas, on doit créer des instantanés en lecture seule.

Pour créer un instantané en lecture seule /mnt/data/.snapshots/web1-version02 du sous-volume /mnt/data/documents/web1 …  
**>> sudo btrfs subvolume snapshot -r /mnt/data/documents/web1 /mnt/data/.snapshots/web1-version02  
Create a readonly snapshot of '/mnt/data/projets/web1' in '/mnt/data/.snapshots/web1-version02'**

Un nouveau sous-volume .snapshots/web1-version02 est créé …  
**>> sudo btrfs subvolume list /mnt/data  
…  
ID 262 gen 33 top level 260 path projets/web1  
ID 263 gen 32 top level 5 path .snapshots/web1-version01  
ID 264 gen 33 top level 5 path .snapshots/web1-version02**

L'instantané /mnt/data/.snapshots/web1-version02 est en lecture seule …  
**>> sudo btrfs subvolume show /mnt/data/.snapshots/web1-version02.snapshots/web1-version02**

**Name: web1-version02**

**UUID: 4f0b67cc-a7a2-554f-979a-568e828e868d**

**Parent UUID: a3eea43b-3c7f-4d4b-b65f-f585b5760eb5**

**Received UUID: -**

**Creation time: 2021-08-27 14:41:32 -0400**

**Subvolume ID: 264**

**Generation: 33**

**Gen at creation: 33**

**Parent ID: 5**

**Top level ID: 5**

**Flags: readonly**

**Snapshot(s):**

On met à jour le fichier index.html à partir du sous-volume /mnt/data/documents/web1 …  
**>> sudo nano /mnt/data/documents/web1/index.html  
<!DOCTYPE html>**

**<html>**

**<head>**

**<!-- En-tête de la page -->**

**<meta charset="utf-8" />**

**<title>Site encore en construction</title>**

**</head>**

**<body>**

**<!-- Corps de la page -->**

**Site en construction -- Disponible fin octobre**

**</body>**

**</html>**

Le fichier index.html du sous-volume /mnt/data/documents/web1 est différent de celui de l’instantané /mnt/data/.snapshots/web1-version02.

**>> cat /mnt/data/documents/web1/index.html  
>> cat /mnt/data/.snapshots/web1-version02/index.html**

On essaie de mettre à jour le fichier index.html de l'instantané /mnt/data/.snapshots/web1-version02 ...  
**>> sudo cp -v /mnt/data/documents/web1/index.html /mnt/data/.snapshots/web1-version02/index.html  
'/mnt/data/documents/web1/index.html' -> '/mnt/data/.snapshots/web1-version02/index.html'  
cp: impossible de créer le fichier standard '/mnt/data/.snapshots/web1-version02/index.html': Système de fichiers accessible en lecture seulement**

Comme on peut le constater, il **n’est pas possible de mettre à jour le fichier index.html** **de l'instantané /**mnt/data/.snapshots/web1-version02 car ce dernier est en **lecture seule**.

## Suppression d'un instantané

Un instantané Btrfs est comme un sous-volume.   
Ainsi, on peut supprimer un instantané Btrfs tout comme on supprime un sous-volume Btrfs.   
On y arrive à l’aide de la même commande.

Voici à quoi ressemble l'arborescence du système de fichiers Btrfs monté sur le répertoire /mnt/data pour le moment …  
**>> tree -a /mnt/data  
/mnt/data**

**documents**

**fichier04**

**fichier05**

**photos**

**fichier08**

**fichier09**

**projets**

**fichier01**

**fichier02**

**fichier03**

**web1**

**index.html**

**style.css**

**.snapshots**

**web1-version01**

**index.html**

**style.css**

**web1-version02**

**index.html**

**style.css**

**videos**

**fichier06**

**fichier07**

**>> sudo btrfs subvolume list /mnt/data  
ID 257 gen 13 top level 5 path photos  
ID 259 gen 13 top level 5 path documents  
ID 260 gen 15 top level 5 path projets  
ID 261 gen 13 top level 5 path videos  
ID 262 gen 36 top level 260 path projets/web1  
ID 263 gen 32 top level 5 path .snapshots/web1-version01  
ID 264 gen 33 top level 5 path .snapshots/web1-version02**

Pour supprimer l'instantané /mnt/data/.snapshots/web1-version02 …  
**>> sudo btrfs subvolume delete /mnt/data/.snapshots/web1-version02  
Delete subvolume (no-commit): '/mnt/data/.snapshots/web1-version02'**

Comme on peut le constater, l'instantané .snapshots/web1-version02 n'est plus …  
**>> sudo btrfs subvolume list /mnt/data  
ID 257 gen 13 top level 5 path photos  
ID 259 gen 13 top level 5 path documents  
ID 260 gen 15 top level 5 path projets  
ID 261 gen 13 top level 5 path videos  
ID 262 gen 36 top level 260 path projets/web1  
ID 263 gen 32 top level 5 path .snapshots/web1-version01**

Comme on peut le constater, les fichiers et répertoires de l'instantané /mnt/data/.snapshots/web1-service02 sont également supprimés …  
**>> tree -a /mnt/data  
/mnt/data**

**documents**

**fichier04**

**fichier05**

**photos**

**fichier08**

**fichier09**

**projets**

**fichier01**

**fichier02**

**fichier03**

**web1**

**index.html**

**style.css**

**.snapshots**

**web1-version01**

**index.html**

**style.css**

**videos**

**fichier06**

**fichier07**

# Autres fonctionnalités

## Compression

Le système de fichiers Btrfs prend en charge la **compression de données au niveau du système de fichiers**.

Cela signifie que les données du système de **fichiers seront compressées automatiquement** au fur et à mesure que de nouvelles données sont **écrites sur le système de fichiers**.   
Lors de l’**accès aux fichiers stockés** sur le système de fichiers, les données de ces **fichiers seront automatiquement** **décompressées**.

Cette fonctionnalité du système de fichiers fera **économiser beaucoup d'espace disque** et permettra de **gagner beaucoup de temps** en comparaison d’une compression manuelle.

**Conditions préalables**

Pour la suite de ce tutoriel …

* Le système de fichiers Btrfs doit être installé sur la station ;
* On doit disposer d'un disque dur ou SSD avec au moins 1 partition libre (peu importe quelle taille).

### Algorithmes de compression btrfs :

Le système de fichiers Btrfs prend en charge les algorithmes de compression suivants …

* **LZO**  
  LZO est un algorithme de compression de blocs en temps réel sans perte.   
  LZO divise les données en blocs et compresse/décompresse les données par blocs en temps réel.   
  C'est l'algorithme de compression par défaut du système de fichiers Btrfs ;
* **ZLIB**  
  ZLIB est une bibliothèque utilisée pour la compression de données.   
  Il utilise l'algorithme de compression de données DEFLATE.   
  L'algorithme de compression de données DEFLATE est une combinaison des algorithmes de codage LZ77 et Huffman.   
  On peut également spécifier le niveau de compression que l’on souhaite.   
  Le niveau peut être n'importe quel nombre de 1 à 9.   
  Un niveau plus élevé indique un taux de compression plus élevé.   
  Ainsi, le niveau 9 économisera plus d'espace disque que le niveau 1 (le niveau 9 a un taux de compression plus élevé que le niveau 1).   
  À moins de spécifier un niveau de compression ZLIB à utiliser, le système de fichiers Btrfs utilisera le niveau de compression ZLIB 3 par défaut ;
* **ZSTD**  
  ZSTD ou ZstanIdard est un algorithme de compression de données sans perte hautes performances.   
  Son taux de compression est comparable à l'algorithme DEFLATE utilisé dans ZLIB, mais est plus rapide.   
  On peut également spécifier le niveau de compression que l’on souhaite.   
  Le niveau peut être n'importe quel nombre de 1 à 15.   
  Un niveau plus élevé indique un taux de compression plus élevé.   
  Ainsi, le niveau 15 économisera plus d'espace disque que le niveau 1 (le niveau 15 a un taux de compression plus élevé que le niveau 1).   
  À moins de spécifier un niveau de compression ZSTD à utiliser, le système de fichiers Btrfs utilisera le niveau de compression ZSTD 3 par défaut.

### Création d'un système de fichiers Btrfs

Pour créer un système de fichiers btrfs avec l’étiquette data sur la partition sdb1 …  
**>> sudo mkfs.btrfs -L data /dev/sdc1**

**btrfs-progs v5.10.1**

**See http://btrfs.wiki.kernel.org for more information.**

**Label: (null)**

**UUID: 2ea04af4-7c3a-44b6-88bb-0b125a6615b8**

**Node size: 16384**

**Sector size: 4096**

**Filesystem size: 18.62GiB**

**Block group profiles:**

**Data: single 8.00MiB**

**Metadata: DUP 256.00MiB**

**System: DUP 8.00MiB**

**SSD detected: no**

**Incompat features: extref, skinny-metadata**

**Runtime features:**

**Checksum: crc32c**

**Number of devices: 1**

**Devices:**

**ID SIZE PATH**

**1 18.62GiB /dev/sdc1**

### Montage du système de fichiers avec compression activée

Pour activer la compression au niveau du système de fichiers, on doit monter le système de fichiers Btrfs avec l'option de montage compress ou compress-force.

* **Option de montage compress**  
  Cette option va simplement activer la compression au niveau du système de fichiers.   
  Le système de fichiers déterminera si la compression rend le fichier à compresser plus gros que la taille du fichier d'origine (non compressé).   
  Si la compression augmente la taille du fichier par rapport à l'original, le système de fichiers ne compressera pas ce fichier ;
* **Option de montage compress-force**   
  Contrairement à l'option de montage compress, si le système de fichiers est monté à l'aide de cette option, alors chaque fichier du système de fichiers sera compressé même si la compression agrandit le fichier.

On crée un répertoire /mnt/data-comp …  
**>> sudo mkdir -v /mnt/data-comp  
mkdir: création du répertoire '/mnt-data-comp'**

Pour activer la compression LZO, on monte le système de fichiers dans le répertoire /mnt/data-comp …  
**>> sudo mount -o compress=lzo /dev/sdc1 /mnt/data-comp**

Pour activer la compression LZO forcée, on monte le système de fichiers dans le répertoire /mnt/data …  
**>> sudo umount /mnt/data-comp  
>> sudo** **mount -o compress-force=lzo /dev/sdc1 /mnt/data**

De la même manière, on peut monter le système de fichiers dans le répertoire /mnt/data-comp comme suit pour activer la compression ZLIB …  
**>> sudo umount /mnt/data-comp  
>> sudo mount -o compress=zlib /dev/sdc1 /mnt/data-comp**

Pour définir un niveau de compression ZLIB (niveau 7), on peut monter le système de fichiers dans le répertoire /mnt/data-comp …  
**>> sudo umount /mnt/data-comp  
>> sudo mount -o compress=zlib:7 /dev/sdc1 /mnt/data-comp**

Pour activer la compression ZSTD, on monte le système de fichiers dans le répertoire /mnt/data-comp …  
**>> sudo umount /mnt/data-comp  
>> sudo mount -o compress=zstd /dev/sdc1 /mnt/data-comp**

Pour définir un niveau de compression ZSTD (niveau 10), on peut monter le système de fichiers dans le répertoire /mnt/data-comp …  
**>> sudo umount /mnt/data-comp  
>> sudo mount -o compress=zstd:10 /dev/sdc1 /mnt/data-comp**

Le système de fichiers que l’on a créé sur la partition sdc1 doit être monté dans le répertoire /mnt/data-comp …  
**>> sudo df -h /mnt/data-comp**

### Montage d'un système de fichiers Btrfs compatible avec la compression au démarrage

Pour **monter le système de fichiers automatiquement au démarrage** avec la compression activée.

On doit d’abord **rechercher l'UUID du système de fichiers Btrfs** pour lequel on aimerait activer la compression …  
**>> sudo btrfs filesystem show /mnt/data  
Label: none uuid: 2ea04af4-7c3a-44b6-88bb-0b125a6615b8  
 Total devices 1 FS bytes used 1.30MiB  
 devid 1 size 18.62GiB used 536.00MiB path /dev/sdc1**

Comme on peut le constater, l'UUID du système de fichiers est **2ea04af4-7c3a-44b6-88bb-0b125a6615b8.**

On accède au fichier /etc/fstab avec l'éditeur de texte nano …  
 **>> sudo nano /etc/fstab**

On entre la ligne suivante à la fin du fichier …  
**UUID=2ea04af4-7c3a-44b6-88bb-0b125a6615b8 /mnt/data-comp btrfs compress=lzo 0 0**

Remarque …  
On remplace l'UUID du système de fichiers Btrfs par le sien.   
On modifie également l'option de montage et l'algorithme de compression selon son choix.

Pour que les modifications prennent effet …À  
**>> sudo mount --all**

On peut constater le point de montage …  
**>> sudo df -h /mnt/data-comp**

### Point de montage systemd

Il est également possible de créer un point de montage systemd.

On peut monter les sous-volumes du système de fichiers Btrfs à l’aide d’un **point de montage systemd** …  
**>> sudo mkdir -v /mnt/compress**  
**>> sudo nano /etc/systemd/system/mnt-compress.mount**

**[Unit]  
Description=Point de montage avec compression**

**[Mount]  
What=/dev/disk/by-uuid/2ea04af4-7c3a-44b6-88bb-0b125a6615b8  
Where=/mnt/compress  
Type=btrfs  
Options=compress=lzo**

**[Install]  
WantedBy=multi-user.target**

**Attention** …  
Le nom du fichier de configuration systemd (**/etc/systemd/system/mnt-compress.mount**) est en fonction du répertoire local (**/mnt/compress**).

Ce nom de fichier systemd doit **obligatoirement** refléter le nom du répertoire.

Il faut par la suite actualiser le démon systemd …

**>> sudo systemctl daemon-reload**

Comme dernière étape, il faut démarrer le point de montage …  
**>> sudo systemctl start mnt-compress.mount**

Pour que le point de montage soit activé à chaque démarrage de la station …

**>> sudo systemctl enable mnt-compress.mount**

**Created symlink /etc/systemd/system/multi-user.target.wants/mnt-compress.mount → /etc/systemd/system/mnt-compress.mount.**

Pour vérifier le point de montage …  
Il faut ensuite renseigner le système d’un nouveau point de montage …   
**>> sudo mount  
…  
/dev/sdc1 on /mnt/compress type btrfs (rw,relatime,space\_cache,subvolid=5,subvol=/)**

**Remarque** …  
Il faut modifier le fichier /etc/fstab en conséquence afin de ne pas monter deux fois le même sous-volume.

# Options pour les points de montage

Le système de fichiers Btrfs possède de nombreuses options de montage que l’on peut utiliser afin de configurer le comportement du système de fichiers (ou des sous-volume) lors de son montage.

## Position des options de montage

Il est possible de monter un système de fichiers Btrfs à l'aide …

* de l’**utilitaire** **mount** (en temps réel);
* du **fichier /etc/fstab** (au démarrage)  
  ou
* d’un point **de montage systemd** (au démarrage).

On peut configurer le comportement du système de fichiers à l'aide des options de montage.

### Ligne de commande

On peut monter un système de fichiers Btrfs à l’aide de la ligne de commande …  
**>> sudo mount -o <Option>,<Option>,<Option>,… <Périphérique ou Sous-volume> <Répertoire>**

Pour **vérifier le point de montage** …  
Il faut ensuite renseigner le système d’un nouveau point de montage …   
**>> sudo mount  
…  
/dev/sdc1 on /mnt/data type btrfs (rw,relatime,space\_cache,subvolid=5,subvol=/)**

### Fichier /etc/fstab

Pour monter le même système de fichiers Btrfs au démarrage à l'aide du fichier /etc/fstab, on doit trouver l'UUID du système de fichiers Btrfs …  
**>> sudo blkid --match-token TYPE=btrfs**   
**/dev/sdb1: UUID="****1d1ade56-fbe6-4ce1-b066-6881d11f264e" UUID\_SUB="fb7cdd8a-5a9e-4b93-a9c5-0d34c2d894d6" BLOCK\_SIZE="4096" TYPE="btrfs" PARTLABEL="btrfs" PARTUUID="ec7ca514-c637-4062-8173-6ce522d3b9e7"**

**/dev/sdc1: UUID="2ea04af4-7c3a-44b6-88bb-0b125a6615b8" UUID\_SUB="0b3fe2f5-cad2-4cdb-8292-7c8b448b9d1c" BLOCK\_SIZE="4096" TYPE="btrfs" PARTLABEL="btrfs" PARTUUID="8ace4066-8958-4540-88f9-a51fb5248ecd"**

On peut alors ajouter au fichier /etc/fstab les informations relatives au point de montage …  
**>> sudo nano /etc/fstab**

Pour monter automatiquement le système de fichiers Btrfs qui a le UUID 1d1ade56-fbe6-4ce1-b066-6881d11f264e sur les répertoire /mnt/data avec les options de montage option1, option2, option3, …  
**UUID=<UUID de la partition> <Répertoire> btrfs <option1>,<option2>,<option3>,… 0 0  
UUID=1d1ade56-fbe6-4ce1-b066-6881d11f264e /mnt/data btrfs option1,option2,option3,… 0 0**

Pour **recharger le fichier /etc/fstab** …  
**>> sudo mount --all**

Pour **vérifier le point de montage** …  
Il faut ensuite renseigner le système d’un nouveau point de montage …   
**>> sudo mount  
…  
/dev/sdc1 on /mnt/data type btrfs (rw,relatime,space\_cache,subvolid=5,subvol=/)**

### Point de montage systemd

On peut monter les sous-volumes du système de fichiers Btrfs à l’aide d’un **point de montage systemd** …  
**>> sudo mkdir -v /mnt/data**  
**>> sudo nano /etc/systemd/system/mnt-data.mount**

**[Unit]  
Description=Point de montage …**

**[Mount]  
What=/dev/disk/by-uuid/<UUID de la partition -- Sous-volume>  
Where=<Répertoire>  
Type=btrfs  
Options=<option1>,<option2>,<option3>,…**

**[Install]  
WantedBy=multi-user.target**

**Description=Point de montage /mnt/data**

**[Mount]  
What=/dev/disk/by-uuid/1d1ade56-fbe6-4ce1-b066-6881d11f264e  
Where=/mnt/data  
Type=btrfs  
Options=compress=lzo**

**[Install]  
WantedBy=multi-user.target**

**Attention** …  
Le nom du fichier de configuration systemd (**/etc/systemd/system/mnt-compress.mount**) est en fonction du répertoire local (**/mnt/compress**).

Ce nom de fichier systemd doit **obligatoirement** refléter le nom du répertoire.

Il faut par la suite actualiser le démon systemd …

**>> sudo systemctl daemon-reload**

Comme dernière étape, il faut démarrer le point de montage …  
**>> sudo systemctl start mnt-data.mount**

Pour que le point de montage soit activé à chaque démarrage de la station …

**>> sudo systemctl enable mnt-data.mount**

**Created symlink /etc/systemd/system/multi-user.target.wants/mnt-data.mount → /etc/systemd/system/mnt-data.mount.**

Pour **vérifier le point de montage** …  
Il faut ensuite renseigner le système d’un nouveau point de montage …   
**>> sudo mount  
…  
/dev/sdc1 on /mnt/data type btrfs (rw,relatime,space\_cache,subvolid=5,subvol=/)**

**Remarque** …  
Il faut modifier le fichier /etc/fstab en conséquence afin de ne pas monter deux fois le même sous-volume.

# Annexe 01 Utilitaires du système de fichiers btrfs

La **commande btrfs filesystem** permet de **gérer** et **générer des rapports** sur les systèmes de fichiers Btrfs.

Les **commandes disponibles** sont …

* **btrfs filesystem df** ;
* **btrfs filesystem show** ;
* **btrfs filesystem sync** ;
* **btrfs filesystem defragment** ;
* **btrfs filesystem resize** ;
* **btrfs filesystem balance** ;
* **btrfs filesystem label**.

## Utilitaire df

Lorsque le système de fichiers est monté et qu'un fichier y est copié, la sortie de la commande df peut produire des informations inexactes pour le système de fichiers Btrfs.

Pour obtenir des informations précises sur l'espace d'un système de fichiers Btrfs, on utilise la commande btrfs filesystem df …  
**>> sudo btrfs filesystem df <Sous-volume>  
>> sudo btrfs filesystem df /mnt/data  
>> sudo btrfs filesystem df /mnt/data/projets  
Data, single: total=32.00MiB, used=0.00B  
System, single: total=4.00MiB, used=16.00KiB  
Metadata, single: total=32.00MiB, used=176.00KiB  
GlobalReserve, single: total=3.25MiB, used=0.00B**

## Utilitaire btrfs filesystem show|sync

On utilise la commande btrfs filesystem show pour afficher la structure d'un système de fichiers.   
La syntaxe est la suivante   
**>> sudo btrfs filesystem show <Options>|<Sous-volume>|<UUID>**

Si on omet le chemin et l'UUID, la commande affiche des informations sur tous les systèmes de fichiers Btrfs   
**>> sudo btrfs filesystem show  
Label: none uuid: 17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e  
 Total devices 1 FS bytes used 192.00KiB  
 devid 1 size 18.62GiB used 68.00MiB path /dev/sdb1**

**Label: none uuid: e4bd01e9-738a-4cdd-b7b1-7659220dff63  
 Total devices 1 FS bytes used 192.00KiB  
 devid 1 size 37.25GiB used 536.00MiB path /dev/sdc1**

On peut utiliser la commande btrfs filesystem sync afin de forcer une synchronisation pour le système de fichiers.   
Le système de fichiers doit être monté.   
Pour forcer une synchronisation du système de fichiers …  
**>> sudo btrfs filesystem sync /mnt/data/projets**

## Utilitaire de défragmentation

Btrfs permet la défragmentation en ligne d'un système de fichiers, d'un fichier ou d'un répertoire. La fonction de

défragmentation en ligne réorganise les données en morceaux contigus dans la mesure du possible afin de créer de plus grandes sections d'espace disque disponible. Ceci permet d’améliorer les performances de lecture et d'écriture.

On utilise la commande btrfs filesystem defragment pour défragmenter un fichier ou un répertoire …  
**>> sudo btrfs filesystem defragment <Commutateur> <Fichier> <Répertoire> [...]**

**Commutateurs**

|  |  |
| --- | --- |
| Commutateur | Description |
| -v ou --verbose | Mode verbeux |
| -c [zlib,lzo,zstd] | Compresser le contenu du fichier lors de la défragmentation |
| -r | Défragmenter les fichiers de manière récursive |
| -f | Vider le système de fichiers après la défragmentation |
| -s start | Défragmenter uniquement à partir du début de l'octet |
| -l len | Défragmenter uniquement jusqu'à len octets |
| -t size | Défragmenter les fichiers au moins en octets |

Il est possible de **configurer la défragmentation automatique** en spécifiant le **commutateur -o autodefrag** lors du montage du système de fichiers.

## Utilitaire btrfs filesystem resize

Btrfs permet de redimensionner à chaud le système de fichiers.   
On utilise la commande btrfs filesystem resize afin de redimensionner un système de fichiers.

Il faut avoir de l'espace disponible pour accueillir le redimensionnement car la commande n'a aucun effet sur les périphériques sous-jacents.

Pour redimensionner un système de fichiers …  
**>> sudo btrfs filesystem resize [devid:][+/-]<[Nouvelle taille> [gkm]|[devid:]max <Sous-volume>**

Descriptions des paramètres …

* **+ <Nouvelle taille>**  
  Augmente la taille du système de fichiers pour la valeur de <Nouvelle taille> ;
* **- <Nouvelle taille>**  
  Diminue la taille du système de fichiers pour la valeur de <Nouvelle taille> ;
* **<Nouvelle taille>**  
  Spécifie la valeur de la nouvelle taille ;
* **g, k ou m**  
  Spécifie l'unité de nouvelle taille (Go, Ko ou Mo)  
  Si aucune unité n'est spécifiée, le paramètre prend par défaut les octets ;
* **max**  
  Spécifie que le système de fichiers occupe tout l'espace disponible

Exemple …  
Pour réduire la taille du système de fichiers de 2 Go …  
**>> sudo btrfs filesystem resize -2G /mnt/data/projets  
Resize '/mnt/data/projets'' of '-2G'**

## Utilitaire btrfs device

On utilise la commande btrfs device afin de gérer les périphériques sur les systèmes de fichiers Btrfs.

La commande btrfs device scan balaie les périphériques physiques à la recherche de membres d'un volume Btrfs.   
Cette commande permet de monter un système de fichiers Btrfs à plusieurs disques sans spécifier tous les disques sur la commande de montage.

Il n’est pas obligé d'exécuter l'analyse de périphérique btrfs à partir de la ligne de commande, car udev exécute automatiquement l'analyse de périphérique btrfs au démarrage.

On utilise la commande btrfs device add pour ajouter un périphérique à un système de fichiers monté …  
**>> sudo btrfs device add /dev/sdc /mnt/data/projets**

On utilise la commande btrfs filesystem balance après avoir ajouté un périphérique …  
**>> sudo btrfs filesystem balance /mnt/data/projets**

On utilise la commande btrfs device delete afin de supprimer un périphérique d'un système de fichiers …  
**>> sudo btrfs device delete /dev/sdc /mnt/data/projets**

## Utilitaire btrfs scrub

On utilise la commande btrfs scrub pour gérer le nettoyage sur les systèmes de fichiers Btrfs.   
Le nettoyage est effectué en arrière-plan par défaut.

La commande btrfs scrub tente de signaler et de réparer les blocs défectueux sur le système de fichiers.

Les commandes disponibles sont …

* **btrfs scrub start** ;
* **btrfs scrub cancel** ;
* **btrfs scrub resume** ;
* **btrfs scrub status**.

On utilise la commande btrfs scrub start pour démarrer un nettoyage sur tous les périphériques d'un système de fichiers ou sur un seul périphérique.  
**>> sudo btrfs scrub start <Sous-volume>  
>> sudo btrfs scrub start /mnt/data/projets  
scrub started on /mnt/data/projets, fsid 17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e (pid=15015)**

On utilise la commande btrfs scrub status pour obtenir l'état d'un travail de scrub.   
L'exemple suivant inclut des informations de nettoyage détaillées sur chaque périphérique du système de fichiers …  
**>> sudo btrfs scrub status -dR <Sous-volume>  
>> sudo btrfs scrub status -dR /mnt/data/projets  
UUID: 17bdf7c0-40a0-42dd-8e6d-dcf583ae5a9e**

**Scrub device /dev/sdc1 (id 1) history  
Scrub started: Wed Aug 25 12:38:02 2021  
Status: finished  
Duration: 0:00:00  
 data\_extents\_scrubbed: 0  
 tree\_extents\_scrubbed: 12  
 data\_bytes\_scrubbed: 0  
 …**

On utilise la commande btrfs scrub cancel pour annuler une tâche de nettoyage en cours …  
**>> sudo btrfs scrub cancel <Sous-volume>  
>> sudo btrfs scrub cancel /mnt/data/projets**

On utilise la commande btrfs scrub resume pour reprendre un nettoyage précédemment annulé ou interrompu …  
**>> sudo btrfs scrub resume <Sous-volume>  
>> sudo btrfs scrub resume /mnt/data/projets**

## Utilitaire btrfs filesyetm label

On utilise la commande btrfs filesystem label pour afficher l’étiquette du système de fichiers …  
**>> sudo btrfs** **filesystem label <Sous-volume>  
>> sudo btrfs filesystem label /mnt/data/projets**

# Annexe 02 Options de montage intéressantes

Les options de montage Btrfs les plus importantes sont …

**acl** et **noacl**  
Les listes de contrôle d’accès (Access Control List ou ACL) gère les autorisations des utilisateurs et des groupes pour les fichiers et répertoires du système de fichiers.

L'option de montage acl active les listes de contrôle d’accès.   
Pour désactiver les listes de contrôle d’accès, on doit utiliser l'option de montage noacl.

Par défaut, les listes de contrôle d’accès sont activées. Ainsi, le système de fichiers utilise l'option de montage acl par défaut.

**autodefrag** et **noautodefrag**La défragmentation d'un système de fichiers Btrfs améliorera les performances du système de fichiers en réduisant la fragmentation des données.

L'option de montage autodefrag permet la défragmentation automatique du système de fichiers.  
L'option de montage noautodefrag désactive cette fonctionnalité.

Par défaut, la défragmentation automatique est désactivée. Ainsi, le système de fichiers Btrfs utilise l' option de montage noautodefrag par défaut.

**compress** et **compress-force**

Cette option contrôle la compression des données au niveau du système de fichiers.  
L’option compress compresse uniquement les fichiers compressé et une taille inférieure aux fichiers originaux   
(si la compression du fichier économise de l' espace disque).

L'option compress-force compresse chaque fichier du système de fichiers même si la compression du fichier augmente sa taille.

Le système de fichiers Btrfs prend en charge de nombreux algorithmes de compression et chacun des algorithmes de compression a différents niveaux de compression.  
Les algorithmes de compression pris en charge par Btrfs sont …

* **lzo** ;
* **zlib** (niveaux 1 à 9)   
  et
* **zstd** (niveaux 1 à 15).

On peut spécifier l'algorithme de compression à utiliser par le système de fichiers avec l'une des options de montage suivantes …

* **compress=<Algorithme>:<Niveau de compression>**
* **compress-force==<Algorithme>:<Niveau de compression>**

**subvol** et **subvolid**

Ces deux options de montage sont utilisées afin de monter séparément un sous-volume spécifique d'un système de fichiers Btrfs.

L'option de montage subvol est utilisée pour monter le sous-volume d'un système de fichiers en utilisant son chemin relatif.  
L'option de montage subvolid est utilisée pour monter le sous-volume d'un système de fichiers Btrfs en utilisant l'ID du sous-volume.

**device**

L'option de montage device est utilisée par le système de fichiers Btrfs multi-périphérique ou le RAID Btrfs.  
Dans certains cas, le système d'exploitation peut ne pas détecter les périphériques de stockage utilisés dans un système de fichiers Btrfs multi-périphériques ou Btrfs RAID.   
Dans de tels cas, on peut utiliser l'option de montage de périphérique pour spécifier les périphériques que l’on souhaite utiliser pour le système de fichiers multi-périphériques Btrfs ou RAID.  
On peut utiliser l'option de montage de périphérique plusieurs fois pour modifier différents périphériques de stockage pour le système de fichiers multi-périphériques Btrfs ou RAID.

On peut utiliser le nom du périphérique (c'est-à-dire sdb , sdc ) ou UUID , UUID\_SUB ou PARTUUID du périphérique de stockage avec l' option de montage de périphérique pour identifier le périphérique de stockage.

Par exemple …

* device=/dev/sdb
* device=/dev/sdb,device=/dev/sdc
* device=UUID\_SUB=490a263d-eb9a-4558-931e-998d4d080c5d
* device=UUID\_SUB=490a263d-eb9a-4558-931e-998d4d080c5d,  
  device=UUID\_SUB=f7ce4875-0874-436a-b47d-3edef66d3424

**commit**

L'option de montage commit est utilisée pour définir l'intervalle (en secondes) dans lequel les données seront écrites sur le périphérique de stockage.

La valeur par défaut est de 30 secondes.

Pour définir l'intervalle de validation à 15 secondes, on peut utiliser l'option de montage commit=15 (disons).

**ssd** et **nossd**

L'option de montage ssd indique au système de fichiers que le système de fichiers utilise un périphérique de stockage SSD, et le système de fichiers effectue l'optimisation SSD nécessaire.  
L'option de montage nossd désactive l'optimisation SSD.

Le système de fichiers détecte automatiquement si un SSD est utilisé.   
Si un SSD est utilisé, l'option de montage ssd est activée.   
Sinon, l'option de montage nossd est activée.

**ssd\_spread** et **nossd\_spread**

L'option de montage ssd\_spread essaie d'allouer de gros morceaux continus d'espace inutilisé à partir du SSD.   
Cette fonctionnalité améliore les performances des SSD bas de gamme.

L'option de montage nossd\_spread désactive la fonctionnalité ssd\_spread.

Le système de fichiers détecte automatiquement si un SSD est utilisé pour le système de fichiers.   
Si un SSD est utilisé, l'option de montage ssd\_spread est activée.   
Sinon, l'option de montage nossd\_spread est activée.

**discard** et **nodiscard**

Si on utilise un SSD supports asynchrones mis en attente TRIM (SATA REV3.1), l’option de montage diuscard permettra au rejet de blocs de fichiers libérés.   
Cela améliorera les performances du SSD.

Si le SSD ne prend pas en charge TRIM asynchrone, l’option de montage discard oeut dégrader les performances du SSD. Dans ce cas, l'option de montage nodiscard doit être utilisée.

Par défaut, l'option de montage nodiscard est utilisée.

**norecovery**

Si l'option de montage norecovery est utilisée, le système de fichiers n'essaiera pas d'effectuer l'opération de récupération des données au moment du montage.

**usebackuproot** et **nousebackuproot**

Si l'option de montage usebackuproot est utilisée, le système de fichiers essaiera de récupérer toute arborescence racine défectueuse ou corrompue au moment du montage.   
Le système de fichiers peut stocker plusieurs racines d'arbre dans le système de fichiers.

L'option de montage usebackuproot recherchera une bonne racine (root) et utilisera la première bonne qu'elle trouvera.

L'option de montage nousebackuproot ne vérifiera pas ou ne récupérera pas les racines incorrectes ou corrompues au moment du montage.   
C'est le comportement par défaut du système de fichiers.

**space\_cache**, **space\_cache=version**, **nospace\_cache** et **clear\_cache**

L'option de montage space\_cache est utilisée pour contrôler le cache d'espace libre.   
Le cache d'espace libre est utilisé pour améliorer les performances de lecture de l'espace libre du groupe de blocs du système de fichiers en mémoire (RAM).

Le système de fichiers prend en charge 2 versions du cache d'espace libre … v1 (par défaut) et v2.

Le mécanisme de mise en cache de l'espace libre v2 améliore les performances des gros systèmes de fichiers   
(taille de plusieurs téraoctets).

On peut utiliser l'option de montage space\_cache=v1 pour définir la v1 du cache d'espace libre et l'option de montage space\_cache=v2 pour définir la v2 du cache d'espace libre.

L'option de montage clear\_cache est utilisée pour effacer le cache d'espace libre.

Lorsque le cache d'espace libre v2 est créé, le cache doit être effacé pour créer un cache d'espace libre v1.

Ainsi, pour utiliser le cache d'espace libre v1 après la création du cache d'espace libre v2, les options de montage clear\_cache et space\_cache=v1 doivent être combinées : clear\_cache,space\_cache=v1

L'option de montage nospace\_cache est utilisée pour désactiver la mise en cache de l'espace libre.

Pour désactiver la mise en cache de l'espace libre après la création du cache v1 ou v2, les options de montage nospace\_cache et clear\_cache doivent être combinées : clear\_cache,nosapce\_cache

**skip\_balance**

Par défaut, l'opération d'équilibrage interrompue ou mise en pause d'un système de fichiers multi-périphériques ou d'un RAID reprendra automatiquement une fois le système de fichiers monté.

Pour désactiver la reprise automatique des opérations d'équilibrage interrompues/en pause sur un système de fichiers multi-périphériques ou un RAID, on peut utiliser l' option de montage skip\_balance.

**datacow** et **nodatacow**

L'option de montage datacow active la fonction Copy-on-Write (CoW) du système de fichiers.   
C'est le comportement par défaut.

Si on souhaite désactiver la fonction de copie sur écriture (CoW) du système de fichiers Btrfs pour les fichiers nouvellement créés, on monte le système de fichiers Btrfs avec l option de montage nodatacow.

**datasum** et **nodatasum**

L'option de montage datasum active la somme de contrôle des données pour les fichiers nouvellement créés du système de fichiers Btrfs.   
Ceci est le comportement par défaut.

Si on ne veut pas que le système de fichiers effectue la somme de contrôle des données pour les fichiers nouvellement créés, on monte le système de fichiers avec l'option de montage nodatasum .