

BTRFS : Installation de Debian dans un subvolume

- Objet : Installer Debian dans un subvolume en BTRFS
- Niveau requis : [avisé,expert](#)
- Commentaires : *Methode d'installation pour installer Debian dans un subvolume BTRFS*
- [Les sauvegardes](#) ; [Le partitionnement](#) ; [Le montage des partitions Les bases du Btrfs](#) 😊
- Suivi : [à-tester](#)
 - Création par [louispolaire](#) 12/03/2014
 - Testé par le
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#) ¹⁾

Ce Tuto est divisé en trois parties :

- [La première partie présente les principes du Btrfs et les commandes de base.](#) À la fin de ce tuto vous serez capable de comprendre la terminologie de Btrfs, de créer un volume et de gérer les subvolumes et snapshots.
- Cette deuxième partie présente l'installation de Debian dans un subvolume. À la fin de ce tuto vous serez capable d'installer Debian dans un subvolume et de modifier Grub pour y booter.
- [La troisième partie présente différentes méthodes pour réaliser des sauvegardes des subvolumes.](#) A la fin de cette partie vous aurez des pistes pour élaborer une stratégie de backup de vos données sur Btrfs.

Ce Tutoriel fait suite à [Btrfs - Les bases](#)

Assurez vous d'avoir bien lu ce précédent tuto avant d'envisager celui-ci !

Installation de / (root) dans un volume Btrfs

Depuis Wheezy, Debian gère le Btrfs lors du processus d'installation. Il y a trois méthodes au moins pour installer Debian sur un système de fichiers Btrfs :

- Utiliser le Btrfs comme un système de fichiers classique et installer Debian à la racine du volume.
 - C'est la méthode la plus simple, mais c'est dommage. L'intérêt de Btrfs c'est de répartir ses données dans différents subvolumes de façon à pouvoir en réaliser des snapshots. Mettre / à la racine du Btrfs ne permet pas cela.
 - Le processus d'installation est le même que pour un autre système de fichiers. Il faudra cependant revoir les options de montages dans le fstab.
 - Il sera éventuellement possible de déplacer tous les fichiers de / dans un subvolume par la suite (avec un `cp -Ra` par exemple). **MAIS ATTENTION !!!** Il faudra apporter des modifications au `grub.cfg` sinon le système ne bootera plus automatiquement. Cet aspect sera traité par la suite.
- Utiliser un méthode d'installation via chroot. La méthode est flexible donc l'installation du système de base directement dans un subvolume sera aisée.
 - C'est la méthode la plus élégante, puisqu'elle est très flexible.
 - Je ne vais pas détailler cette méthode ici, il y a des tutoriels bien faits ([par exemple ici](#)).

Sachez juste qu'il suffit au moment du montage des partitions d'ajouter dans les options de montage **subvol=chemin/vers/le/subvolume/racine** pour utiliser le subvolume comme racine.

- Grub sait maintenant bien gérer le Btrfs même à l'intérieur d'un subvolume. Par contre, dans le cas d'un multi-boot, os-prober ne va pas fouiller dans tous les subvolumes, donc votre nouveau système ne saura pas trouvé par la commande update-grub d'un autre système. [Je propose une méthode plus bas.](#)
- La méthode d'installation via chroot est utilisée par Arch et Gentoo entre autres. Il est donc simple d'installer ces distributions dans un subvolume.
- Utiliser le mode d'installation expert pour installer Debian dans un subvolume.
 - La méthode est moins élégante que la précédente et c'est pour ça que je vais la détailler. 😊
 - Il est possible d'utiliser un cd d'installation de Wheezy, même si, encore une fois, il vaut mieux avoir un kernel à jour. Rien ne vous empêche d'activer les dépôts de testing ou sid après installation.

Installation dans un subvolume à l'aide du cd d'installation Debian

On se place ici dans le cas où vous avez un système déjà bootable (par exemple Debian sur une partition ext4 qui boote avec Grub). Je suppose que vous avez de l'espace sur votre disque pour créer une partition de 20Go environs en Btrfs. (le but étant de tester)

Si vous utilisez un disque vierge, c'est pareil au niveau de l'installation, par contre Grub ne gère pas les disques Btrfs sans table de partition (mkfs.Btrfs /dev/sda). Il vous faut créer un table des partitions avec un MBR et juste une partition primaire sur tout le disque (ou avec un swap aussi si vous voulez).

Donc :

- Pour un disque bootable il faut une table des partitions (MBR, GPT..) et donc le volume Btrfs sera du type /dev/sdaX
- Pour un disque de donnée vous pouvez vous passer de la table de partitions, le volume Btrfs sera de type /dev/sda, mais le disque ne sera pas bootable seul. Rien ne vous empêche cependant de booter depuis un autre disque. Par contre l'installateur de Debian ne fonctionnera pas sur ce type de disque, il faudra passer pas la méthode chroot.

Dans le cas où il y aurait déjà des données ou un système d'exploitation dans le volume Btrfs cette méthode marche aussi.

Rebootez sur le CD ou l'USB d'installation.

Allez dans les Options Avancées et choisissez l'Installation Expert (graphique ou pas)

Faites l'installation comme normalement jusqu'au partitionnement.

```
Choisir la langue
Configurer le clavier
Détecer et monter le CD
Charger des composants d'installation à partir du CD
Détecer le réseau
Configurer le réseau
Créer les utilisateurs et choisir les mots de passe
```

```
Configurer l'horloge  
Détecter les Disques
```

Une fois les disques détectés :

```
Exécuter un shell (ligne de commande)
```

On monte le volume Btrfs dans /mnt

```
mount -t btrfs -o ..options.. /dev/sdaX /mnt
```

On va créer les subvolumes où l'on désire installer Debian :

```
cd /mnt
```

```
btrfs subvolume create debian
```

```
cd debian
```

```
btrfs subvolume create root
```

```
btrfs subvolume create home
```

Si vous êtes adepte d'un /usr séparé ajoutez le subvolume correspondant. Après c'est une histoire d'organisation.

Étant donné que les snapshots ne sont pas récursifs il peut être intéressant de mettre /proc et /var dans des subvolumes afin de limiter la taille des snapshots.

On monte les subvolumes dans /target :

```
cd /
```

```
mkdir -p /target
```

```
mount -t btrfs -o ..options..,subvol=debian/root /dev/sdaX /target
```

```
mkdir /target/home
```

```
mount -t btrfs -o ..options..,subvol=debian/home /dev/sdaX /target/home
```

```
(umount /mnt)
```

```
cat /proc/self/mountinfo
```

```
exit
```

La dernière commande devrait vous donner la liste des volumes montés y compris les subvolumes.

Et on se lance pour l'installation du système de base (qui va s'installer dans /target donc...)

Installer le système de base

Normalement l'installateur va se plaindre que vous n'êtes pas passés par l'étape "Partitionner les disques" Choisissez "Revenir en arrière" jusqu'à qu'il lâche l'affaire (2 fois) et installe le système de base.

Et voilà ! le système s'installe et le reste de l'installation peut se faire normalement.

GRUB

Grub est maintenant capable d'ajouter lui même les options de montage dans grub.cfg. Donc si vous installez un système tout neuf il n'y aura pas de soucis, il s'occupera de détecter que le /boot est dans un subvolume, écrira dans le MBR et tout ira bien.

Grub est même capable de booter depuis un /boot compressé en lzo ou zlib !

Ce n'est pas le cas de Syslinux. Pour utiliser syslinux il vous faudra empêcher la compression de /boot :

```
cd /subvolume/racine/
```

```
mv /boot /boot-old
```

```
mkdir /boot
```

```
chattr -c /boot
```

```
cp --reflink -R /boot-old/* /boot/
```

```
rm -R /boot-old
```

L'option **-reflink** va forcer la duplication des données sur le disque lors de la réécriture dans le /boot sans compression.

Ceci dit je vais quand même expliquer ici comment configurer le grub.cfg, pour tous les autres cas et au cas où vous fassiez un erreur.

ATTENTION assurez vous d'avoir une solution de secours **AVANT** d'installer votre système.

Personne n'est à l'abri d'une faute de frappe.

Et une faute de frappe peut rendre votre système non-bootable.

Donc **munissez vous d'un LiveCD ou d'un LiveUSB** afin de pouvoir avoir accès à votre grub.cfg en cas de pb.

Avant de vous lancer soyez sur que vous avez bien lu les tutoriels suivants :

- [Grub2 vision d'ensemble](#)
- [Grub2 l'installation](#)
- [Scripts manuels](#)

Dans le cas ou vous voudriez que votre "vieux" système continue de booter depuis le mbr et que son

grub.cfg s'occupe de faire booter le "nouveau" système sur Btrfs :

Toujours dans l'installateur Debian installez grub. **Mais n'installez pas le programme de démarrage GRUB sur le secteur d'amorçage !**

Vous pourrez désinstaller grub une fois que vous aurez booté le "nouveau" système.
Vous pouvez aussi continuer sans programme d'amorçage et utiliser le modèle de fichier 40_custom donné ci dessous **à vos risques et périls**

```
Installer le programme de démarrage GRUB sur le secteur d'amorçage ?
-> Non
puis..
Périphérique où sera installé le programme de démarrage
Laissez le champ vide
```

De cette façon on va pouvoir utiliser directement la partie concernant le boot du "nouveau" système du grub.cfg dans le "vieux" grub.cfg.

```
Terminer l'installation
```

Redémarrez votre "vieux" système

Ouvrez le grub.cfg du nouveau système :

```
mount /dev/sdaX -o ..options.. /mnt
```

```
cd /mnt/debian/root/boot/grub
```

```
mousepad grub.cfg
```

Copiez la partie après ### BEGIN ...

[extrait grub.cfg](#)

```
menuentry 'Debian GNU/Linux, avec Linux 3.2.0-4-amd64' --class debian -
-class gnu-linux --class gnu --class os {
    load_video
    insmod gzio
    insmod part_gpt
    insmod Btrfs
    set root='(hd0,gpt3)'
    search --no-floppy --fs-uuid --set=root
9201243d-1559-4b27-8973-070aeefa1079
    echo    'Chargement de Linux 3.2.0-4-amd64 ...'
    linux   /debian/root/boot/vmlinuz-3.2.0-4-amd64
root=UUID=9201243d-1559-4b27-8973-070aeefa1079 ro
rootflags=subvol=debian/root quiet
    echo    'Chargement du disque mémoire initial ...'
    initrd  /debian/root/boot/initrd.img-3.2.0-4-amd64
}
```

Il faut que vous pensiez à modifier/vérifier les paramètres suivants :

- set root='(hd0,gpt3)' pour adapter au numéro de votre partition
- les UUIDs
- il faut indiquer le chemin vers vmlinuz et initrd depuis la racine du volume Btrfs (/debian/root/boot/vmlinuz...)
- ajouter une option de boot : **rootflags=subvol=chemin-vers-la-racine**

Si vous n'avez pas installé Grub et n'avez donc pas de grub.cfg vous pouvez utiliser le schéma ci dessus en prenant soin de l'adapter à votre système

Ouvrez 40_custom de votre "vieux" système et collez y le menuentry précédent dûment modifié. Attention à ne pas effacer les premières lignes !

```
nano /etc/grub.d/40_custom
```

40_custom

```
#!/bin/sh
exec tail -n +3 $0
# This file provides an easy way to add custom menu entries.  Simply
# type the
# menu entries you want to add after this comment.  Be careful not to
# change
# the 'exec tail' line above.
menuentry 'Debian GNU/Linux, avec Linux 3.2.0-4-amd64' --class debian -
-class gnu-linux --class gnu --class os {
    load_video
    insmod gzio
    insmod part_gpt
    insmod Btrfs
    set root='(hd0,gpt3)'
    search --no-floppy --fs-uuid --set=root
9201243d-1559-4b27-8973-070aeefa1079
    echo    'Chargement de Linux 3.2.0-4-amd64 ...'
    linux   /debian/root/boot/vmlinuz-3.2.0-4-amd64
root=UUID=9201243d-1559-4b27-8973-070aeefa1079 ro
rootflags=subvol=debian/root quiet
    echo    'Chargement du disque mémoire initial ...'
    initrd  /debian/root/boot/initrd.img-3.2.0-4-amd64
}
```

```
chmod +x /etc/grub.d/40_custom
```

```
update-grub
```

Profitez en pour éditer le fstab du nouveau système :

```
nano /mnt/debian/root/etc/fstab
```

```
#l'entrée pour la racine
UUID=a69d9182-f4c7-4276-b35d-7d5f9bd50a57      /      Btrfs
rw,noatime,ssd,discard,
autodefrag,subvol=debian/root      0      0
#l'entrée pour le home
UUID=a69d9182-f4c7-4276-b35d-7d5f9bd50a57      /home      Btrfs
rw,noatime,ssd,discard,
autodefrag,subvol=debian/home      0      0
#swap et tmpfs
swap...
tmpfs /tmp tmpfs default      0      0
#Je vous conseille de monter le volume Btrfs à la racine quelque part afin
d'y avoir accès
facilement
UUID=a69d9182-f4c7-4276-b35d-7d5f9bd50a57      /home/Btrfs      Btrfs
rw,noatime,ssd,discard,
autodefrag,nodev,nosuid,noexec,subvolid=0      0      0
```

Et voilà ! Au redémarrage vous aurez une entrée "Debian GNU/Linux, avec Linux 3.2.0-4-amd64" dans votre menu Grub et ça devrait booter !!

Références

- [Le wiki de Btrfs](#)
- [La page sur Btrfs sur le wiki d'Archlinux](#)
- [\[\[http://zythmer.acyclic.org/blog/2013/01/27/debian-full-disk-encryption-btrfs-subvolumes.html|http://zythmer.acyclic.org/blog/2013/01/27/debian-full-disk-encryption-btrfs-subvolumes.html\]\]](http://zythmer.acyclic.org/blog/2013/01/27/debian-full-disk-encryption-btrfs-subvolumes.html) - **Lien Obsolète**

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

From:
<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:
<http://debian-facile.org/doc:systeme:btrfs-root-install-subvol>

Last update: **28/10/2015 18:45**

