

B-Tree Filesystem (BTRFS)

黃柏竣 (bjhuang)

Introduction

- BTRFS is a modern copy on write (CoW) filesystem for Linux aimed at implementing advanced features while also focusing on fault tolerance, repair and easy administration.
 - Snapshots which do not make a full copy of the files
 - Built-in volume management, support for software-based RAID 0, RAID 1 and others
 - Self-healing - checksums for data and metadata, automatic detection of silent data corruptions
 - Transparent data compression
 - Reflinks, fast and efficient file copies

```
# mount -o compress=zstd /dev/sdx /mnt  
  
# cp --reflink=always source target
```

Why Btrfs?

- ext4 is the most common filesystem with basic functions.
- Btrfs has other capabilities, such as snapshots.
 - ZFS also has many capabilities while utilizing relatively high memory.

Subvolume

- A BTRFS subvolume is a part of filesystem with its own independent file/directory hierarchy and inode number **namespace**.
- Each Btrfs file system has a **top-level** subvolume with ID 5
 - The top-level subvolume has path / on the file system
 - Other subvolumes are nested below the top-level subvolume
 - By default, the top-level subvolume is mounted when mounting the file system
- Naming convention
 - name with “@” prefix
 - @ for /
 - @www for /var/www

Nested layout

- A subvolume is created inside another subvolume.
- You need to delete all children subvolumes before deleting the parent one.
- Subvolumes **don't** need to be mounted manually.
 - For each of these subvolumes the mount options of their mountpoint applies.

```
toplevel          (volume root directory, not mounted)
  \-- @           (subvolume root directory, to be mounted at /)
    +-- @home     (subvolume root directory)
    +-- @var       (subvolume root directory)
      +-- @www     (subvolume root directory)
      +-- lib      (directory)
        \-- @postgresql (subvolume root directory)
```

Flat layout (1/2)

- All subvolumes are top level's children.
- All subvolumes need to be mounted manually.
 - Each of these subvolumes/mountpoints can be mounted with different options.

```
toplevel      (volume root directory, not to be mounted by default)
+-- @         (subvolume root directory, to be mounted at /)
+-- @home     (subvolume root directory, to be mounted at /home)
+-- var       (directory)
|   |-- @www  (subvolume root directory, to be mounted at /var/www)
|-- @postgres (subvolume root directory, to be mounted at /var/lib/postgresql)
```

Flat layout (2/2)

- /etc/fstab may look like below:

```
UUID=the-btrfs-fs-uuid    /                btrfs subvol=@,defaults,autodefrag    0 0
UUID=the-btrfs-fs-uuid    /home            btrfs subvol=@home,defaults,noatime    0 0
UUID=the-btrfs-fs-uuid    /var/www         btrfs subvol=/var/@www,noatime         0 0
UUID=the-btrfs-fs-uuid    /var/lib/postgresql btrfs subvol=/var/@postgres,noatime    0 0
```

Snapshot

- A snapshot is simply a subvolume that shares its data (and metadata) with some other subvolume, using Btrfs's **COW** capabilities.
 - Not recursive
 - -r option for read-only
 - There is no difference between the original subvolume and writable snapshot

```
btrfs subvolume snapshot [-r] <source subvolume> <destination subvolume>
```


LVM snapshot v.s. Btrfs snapshot

- Common
 - Copy on write
- LVM snapshot
 - Need to specify LV size when creating a snapshot
 - Write all snapshot volumes for same LV
 - slow write speed and waste more space
- Btrfs snapshot
 - No need to specify size for snapshot
 - Multiple snapshots (reflink) won't waste performance & space

Snapshot rollback

- The official documentation is unclear on the procedure for performing snapshot rollback.
 - Mount the snapshot subvolume and copy files to your destination folder
 - Snapshot the snapshot subvolume to another subvolume writable and remount
 - ...

```
root@trixws:~# btrfs subvolume list -t /mnt
ID      gen      top level    path
--      ---      -
261     47       5            home/@ray
267     39       5            snapshots/@homeray-1762502638

root@trixws:~# btrfs subvolume snapshot /mnt/snapshots/@homeray-1762502638 /mnt/home/@ray_new

root@trixws:~# mount -o subvol=/home/@ray_new /dev/sdb /home/ray
```

Example

- Migrate home directory from ext4 to btrfs

Migrate home directory & snapshot

1. Install btrfs for your OS (`apt install btrfs-progs` on Debian)
2. Add a block device (partition/disk or LV, ...) for btrfs
3. `mkfs.btrfs` for the new block device
4. Mount to temporary location and create btrfs subvolume for home
5. Copy home directory to the new subvolume
6. Mount new subvolume to `/home/USER`
7. Create snapshot subvolume for home subvolume

Subvolume layout

```
toplevel      (volume root directory, not to be mounted by default)
  +-- home     (directory)
    |-- @USER  (subvolume root directory, to be mounted at /home/USER)
  +-- snapshots (directory)
    |-- @homeUSER (subvolume root directory, for storing /home/USER snapshot)
```

Migrate home directory (LV)

```
root@trixws:~# lvs
```

LV	VG	Attr	LSize	Pool	Origin	Data%	Meta%	Move	Log	Cpy%	Sync	Convert
home-vol	home-vg	-wi-a-----	3.99g									
root	trixws-vg	-wi-ao----	<44.86g									
swap_1	trixws-vg	-wi-ao----	<1.23g									

```
root@trixws:~# lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINTS
sda	8:0	0	4G	0	disk	
├─sda1	8:1	0	2G	0	part	
├─┬─home--vg-home--vol	254:2	0	4G	0	lvm	
└─sda2	8:2	0	2G	0	part	
└─┬─home--vg-home--vol	254:2	0	4G	0	lvm	
sdb	8:16	0	48G	0	disk	
├─sdb1	8:17	0	976M	0	part	/boot/efi
├─sdb2	8:18	0	977M	0	part	/boot
├─sdb3	8:19	0	24G	0	part	
├─┬─trixws--vg-root	254:0	0	44.9G	0	lvm	/
├─┬─trixws--vg-swap_1	254:1	0	1.2G	0	lvm	[SWAP]
└─sdb4	8:20	0	22.1G	0	part	
└─┬─trixws--vg-root	254:0	0	44.9G	0	lvm	/

Migrate home directory (mkfs)

- Use `btrfs filesystem show` to show btrfs structure

```
root@trixws:~# mkfs.btrfs /dev/home-vg/home-vol
btrfs-progs v6.14
See https://btrfs.readthedocs.io for more information.

Performing full device TRIM /dev/home-vg/home-vol (3.99GiB) ...
...
Number of devices: 1
Devices:
  ID        SIZE  PATH
  1         3.99GiB /dev/home-vg/home-vol

root@trixws:~# btrfs filesystem show
Label: none  uuid: 0ed693ae-28ac-44ce-8674-9059f685d8f1
Total devices 1 FS bytes used 144.00KiB
devid    1 size 3.99GiB used 536.00MiB path /dev/mapper/home--vg-home--vol
```

Migrate home directory (subvolume & copy)

```
root@trixws:~# mount /dev/home-vg/home-vol /mnt  
  
root@trixws:~# mkdir -p /mnt/home  
  
root@trixws:~# btrfs subvolume create /mnt/home/@ray  
Create subvolume '/mnt/home/@ray'  
  
root@trixws:~# cp -a /home/ray/. /mnt/home/@ray/
```


Migrate home directory (mount)

```
root@trixws:~# mount -o subvol=/home/@ray /dev/home-vg/home-vol /home/ray

root@trixws:~# df -h
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/trixws--vg-root 44G   1.2G   41G   3% /
...
/dev/mapper/home--vg-home--vol 4.0G   8.6M   3.5G   1% /mnt
/dev/mapper/home--vg-home--vol 4.0G   8.6M   3.5G   1% /home/ray

# ensure consistency after reboot
root@trixws:~# echo "UUID=0ed693ae-28ac-44ce-8674-9059f685d8f1 /home/ray btrfs subvol=/home/@ray 0 0" >> /etc/fstab
```

Migrate home directory (snapshot)

```
root@trixws:~# mkdir -p /mnt/snapshots

root@trixws:~# btrfs subvolume snapshot /mnt/home/@ray /mnt/snapshots/@homeray-$(date +%s)
Create subvolume '/mnt/snapshots/@homeray-1762502638'

root@trixws:~# btrfs subvolume list -t /mnt
```

ID	gen	top level	path
--	---	-----	----
261	47	5	home/@ray
267	39	5	snapshots/@homeray-1762502638

Reference

- [Welcome to BTRFS documentation! — BTRFS documentation](#)
- [SysadminGuide - btrfs](#)
- [Btrfs - ArchWiki](#)
- [Btrfs migration - Debian Wiki](#)
- [Overview of file systems in Linux | Storage Administration Guide | SLES 15 SP7](#)