Understanding Linux file systems

Linux supports a number of different file systems with different sets of features and intended use-cases.

Ext4: The All-rounder

Ext4 is the latest iteration of the "Extended file system" and the default on most Linux distributions. It supports journaling, which means the file system keeps a list of files that are to be written to the disk and once the file has been written, it is removed from the journal. This improves file system integrity in case of a power loss. It also features delayed allocation, which aims to improve flash memory life. Ext4 also actively prevents file fragmentation when writing data.

Btrfs: The new kid on the block

Btrfs is a new type of Linux file system that is designed differently from Ext4 in some respects.

Btrfs is a copy-on-write (CoW for short) file system, which means that copies of files are only "virtual" and do not occupy any additional storage space, and a copy only becomes "real" once it has been changed. Writes do not overwrite data in place; instead, a modified copy of the block is written to a new location, and metadata is updated to point at the new location.

Btrfs organizes its data in subvolumes, which can be mounted like partitions. Unlike partitions, subvolumes do not have a fixed size. Instead, subvolumes are merely an organizational unit on the same Btrfs partition, the size of which depends on the contents stored in them. Any number of subvolumes can be created for different mount points, e.g. `/` and `/home`. This allows, amongst other things, for multiple operating systems to be installed to the same disk on the same computer without interfering with each other.

Another feature of Btrfs is its ability to create snapshots of the file system. The state of a subvolume can be recorded in a snapshot, e.g. before a critical system update, in order to revert to a previous state of the file system if necessary. Thanks to CoW, snapshots require very little storage space compared to full-fledged backups (although they are no replacement for them!) and can be mounted and booted from like regular subvolumes. This makes it possible to "rewind" the

state of the file system with comparatively little effort. Tools such as `snapper` or `timeshift` can simplify and automate the process of creating during system updates and restoring from snapshots from the commandline.

Btrfs also implements transparent compression of data blocks. Written data is automatically stored in compressed form if the appropriate mount options are set. There are a number of different compression algorithms to choose from, including lz4, gzip and Zstandard. This can also increase the life span of flash based storage devices, as less data is written to the disk and not as much wear-leveling is taking place.

Btrfs comes with RAID management for RAID 0, 1 and 10 built into the file system itself and makes an additional software or firmware RAID superfluous for these configurations. In addition, the integrated RAID functionality offers the advantage that it is aware of used and free data blocks in mirrored setups, which can considerably speed up the reconstruction of a RAID, as only the used blocks are reconstructed. Using the built-in RAID functionality in Btrfs also allows for more storage devices to be added to the RAID later on.

XFS: large data made easy

XFS is a high-performance file system particularly proficient at parallel I/O due to its allocation group based design. This makes it ideal for when you're dealing with bandwidth intensive tasks, i.e. multiple processes accessing the file system simultaneously. Like ext4 it contains a journal for file system consistency.

XFS keeps an overview over the free space on the file system, allowing it to quickly determine free blocks large enough for new data in order to prevent file fragmentation.

Revision #3
Created 17 December 2023 12:36:35 by Sebin
Updated 23 December 2023 20:00:30 by Sebin