BTRFS
stable

Search docs

OVERVIEW
Introduction
Status
Manual pages
btrfs(8)
btrfs(5)
btrfs-balance(8)
btrfs-check(8)
btrfs-convert(8)
btrfs-device(8)
btrfs-filesystem(8)
btrfs-find-root(8)
btrfs-image(8)
btrfs-inspect-internal(8)
btrfs-ioctl(2)
btrfs-map-logical(8)
btrfs-property(8)
btrfs-qgroup(8)
btrfs-quota(8)
btrfs-receive(8)
SYNOPSIS
DESCRIPTION
BUGS
EXIT STATUS
AVAILABILITY
SEE ALSO
btrfs-replace(8)
btrfs-rescue(8)
btrfs-restore(8)
btrfs-scrub(8)
btrfs-select-super(8)
btrfs-send(8)
btrfs-subvolume(8)
btrfstune(8)
fsck.btrfs(8)
mkfs.btrfs(8)
Administration
Hardware considerations
Changes (feature/version)
Changes (kernel/version)
Changes (btrfs-progs)
Contributors
Glossary
Installation instructions
Source repositories
Interoperability
FEATURES
Common Linux features
Custom ioctls
Auto-repair on read
Balance
Compression
Checksumming
Convert
Deduplication
Defragmentation
Inline files
Quota groups
Reflink
Resize
Scrub
Seeding device
Send/receive
Subpage support
Subvolumes
Swapfile
Tree checker
Trim/discard
Volume management
Zoned mode
DEVELOPER DOCUMENTATION
Development notes
Developer's FAQ
Conventions and style for documentation
Experimental features
Btrfs design
Btrees
On-disk Format
Send stream format
JSON output
Internal APIs
Release checklist
Pull request review workflow
Command line, formatting, UI guidelines
btrfs-ioctl(2)
TODO

btrfs-receive(8)

SYNOPSIS

btrfs receive [options] <path>

or

btrfs receive --dump [options]

DESCRIPTION

Receive a stream of changes and replicate one or more subvolumes that were previously generated by **btrfs send**. The received subvolumes are stored to *path*, unless *--dump* option is given.

If *--dump* option is specified, **btrfs receive** will only do the validation of the stream, and print the stream metadata, one operation per line.

btrfs receive will fail in the following cases:

1. receiving subvolume already exists
2. previously received subvolume has been changed after it was received
3. default subvolume has changed or you didn't mount the filesystem at the toplevel subvolume

A subvolume is made read-only after the receiving process finishes successfully (see BUGS below).

`Options`

`-f` *<FILE>*

read the stream from *FILE* instead of stdin,

-C|--chroot

confine the process to *path* using chroot(1)

`-e`

terminate after receiving an *end cmd* marker in the stream.

Without this option the receiver side terminates only in case of an error on end of file.

-E|--max-errors <NERR>

terminate as soon as NERR errors occur while stream processing commands from the stream

Default value is 1. A value of 0 means no limit.

`-m` *<ROOTMOUNT>*

the root mount point of the destination filesystem

By default the mount point is searched in :file:/proc/self/mounts`. If `/proc` is not accessible, e.g. in a chroot environment, use this option to tell us where this filesystem is mounted.

`--force-decompress`

if the stream contains compressed data (see *--compressed-data* in btrfs-send(8)), always decompress it instead of writing it with encoded I/O

`--dump`

dump the stream metadata, one line per operation

Does not require the *path* parameter. The filesystem remains unchanged. Each stream command is on one line in the form of *key=value* and separated by one or more spaces. Values that contain special characters (like paths or extended attributes) are encoded in C-like way, e.g. '*\n*' or octal escape sequence like '*\NNN*' where N is the char value. Same encoding as is used in */proc* files.

-q|--quiet

(deprecated) alias for global *-q* option

`-v`

(deprecated) alias for global *-v* option

`Global options`

-v|--verbose

increase verbosity about performed actions, print details about each operation

-q|--quiet

suppress all messages except errors

BUGS

btrfs receive sets the subvolume read-only after it completes successfully. However, while the receive is in progress, users who have write access to files or directories in the receiving *path* can add, remove, or modify files, in which case the resulting read-only subvolume will not be an exact copy of the sent subvolume.

If the intention is to create an exact copy, the receiving *path* should be protected from access by users until the receive operation has completed and the subvolume is set to read-only.

Additionally, receive does not currently do a very good job of validating that an incremental send stream actually makes sense, and it is thus possible for a specially crafted send stream to create a subvolume with reflinks to arbitrary files in the same filesystem. Because of this, users are advised to not use *btrfs receive* on send streams from untrusted sources, and to protect trusted streams when sending them across untrusted networks.

EXIT STATUS

btrfs receive returns a zero exit status if it succeeds. Non zero is returned in case of failure.

AVAILABILITY

btrfs is part of btrfs-progs. Please refer to the documentation at https://btrfs.readthedocs.io.

SEE ALSO

btrfs-send(8), mkfs.btrfs(8)

Previous Next

stable