

btrfs is awesome

except when it isn't

Richard Brown
openSUSE Chairman
btrfs fan
rbrown@opensuse.org

I ♥ btrfs

- Snapshots & Rollback
- Transactional Updates
- Send/Receive
 - https://btrfs.wiki.kernel.org/index.php/Incremental_Backup



I ♥ btrfs

- Compression
 - `mount -o compress /dev/sdx /mnt`
 - `fstab: UUID=1a2b3c4d /home btrfs subvol=@/home,compress 0 0`
 - Existing files can be encrypted with ``btrfs filesystem defrag -r /path``
- 3 different compression methods
 - `zlib` (Slow, High ratio)
 - `lzo` (Fast, Low ratio)
 - `zstd` (Fast, High ratio, New)





The background features abstract geometric shapes in two shades of green. A large, dark teal shape occupies the left and top-left portions of the frame. To its right, a lighter green shape is partially visible, separated by a white border. The overall composition is clean and modern.

Or is it?

How much space is being used?

- btrfs snapshots complicate the calculations of disk space use
- As snapshots only contain diffed blocks, a full, accurate calculation would require the equivalent of `du` checking every file in every snapshot
- `df` does not do this





Space in use:

- Do not use `df` on btrfs
- Use one of the following instead (ordered by detail)
 - `btrfs filesystem show /`
 - `btrfs filesystem df /`
 - `btrfs filesystem usage /`





Out of Space

- It's probably not btrfs fault
- Blame snapper!





Snapper is better now

- No more timeline snapshots by default
- Space-aware cleanup
 - Default on new installations
 - `snapper setup-quota` needed on older installs
 - <http://snapper.io/2016/05/18/space-aware-cleanup.html>



Tidying up snapper snapshots

- Relocate data in empty/near empty data chunks to free up enough space to delete again
 - `btrfs fi balance start </mountpoint> -dusage=5`
- Remove snapper snapshots
 - `snapper -c root list`
 - `snapper -c root delete snapshot_number(s)`







Don't btrfsck/btrfs check --repair



Instead

- `btrfs scrub start /dev/sdX`
 - Monitor with ``btrfs scrub status /dev/sdX``
- Attempt booting with backup btrfs root tree
 - `mount -o usebackuproot /dev/sdaX /mnt`



If that didn't fix it

- Most likely a HW problem!
 - An offline disk **will** cause RO file system!
- Run `btrfs check`
 - **Not** --repair
 - Store the logs for the bug report – <https://bugzilla.opensuse.org>
- Backup/recover all data to a second device
 - `btrfs restore /dev/sda1 /mnt/usbdrive`



Mostly Harmless Rescue Options

- `btrfs rescue super-recover /dev/sdaX`
- `btrfs rescue zero-log /dev/sdaX`
- `btrfs rescue fix-device-size /dev/sdaX`
- `btrfs rescue chunk-recover /dev/sdX`
 - SLOW



Still not working?

- Check the HW



Still not working?

- Check the HW
- Make a file backup



Still not working?

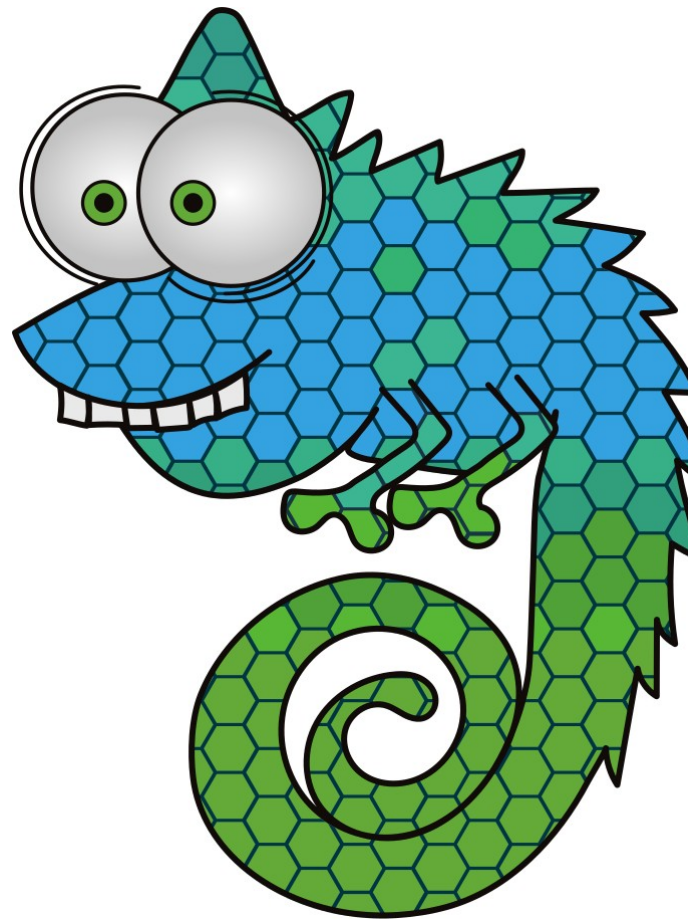
- Check the HW
- Make a file backup
- Backup or use btrfs restore



Still not working?

- Check the HW
- Make a file backup
- Backup or use btrfs restore
- THEN consider using `btrfs check --repair``





Join Us at www.opensuse.org



License

This slide deck is licensed under the Creative Commons Attribution-ShareAlike 4.0 International license. It can be shared and adapted for any purpose (even commercially) as long as Attribution is given and any derivative work is distributed under the same license.

Details can be found at <https://creativecommons.org/licenses/by-sa/4.0/>

General Disclaimer

This document is not to be construed as a promise by any participating organisation to develop, deliver, or market a product. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. openSUSE makes no representations or warranties with respect to the contents of this document, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. The development, release, and timing of features or functionality described for openSUSE products remains at the sole discretion of openSUSE. Further, openSUSE reserves the right to revise this document and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes. All openSUSE marks referenced in this presentation are trademarks or registered trademarks of SUSE LLC, in the United States and other countries. All third-party trademarks are the property of their respective owners.

Credits

Template
Richard Brown
rbrown@opensuse.org

Design & Inspiration
openSUSE Design Team
<http://opensuse.github.io/branding-guidelines/>

Lab Time

- Installation
- Snapper
- btrfs check
- Booting read only snapshot

Installation

- Btrfs default root fs for SuSE – other distros?
- Subvolumes automatically added if root partition
- Review/prune subvolumes for accuracy

Snapper

- Tool for managing snapshots
- Snapshot types:
 - Single
 - *Pre update*
 - *Post update*
- Snapshots saved in `/.snapper`

.snapper structure

- Snapshots are saved in /.snapper
 - 1/ 17/ 18/ 2/ 80/ 81/ 82/ 83/ 84/ 85/ 86/ 87/ 88/ 89/ 90/ 91/ grub-snapshot.cfg
- # ls /.snapshots/1
 - info.xml snapshot/
- # ls /.snapshots/1/
 - info.xml snapshot/
- # ls /.snapshots/1/snapshot/
 - bin/ boot/ dev/ etc/ home/ lib/ lib64/ mnt/ opt/ proc/ root/ run/ sbin/ srv/ sys/ tmp/ u/ usr/ var/

File system info

- # btrfs fi show

```
Label: none  uuid: b5035aed-d0bb-4c19-9243-e313e57fec37
Total devices 1 FS bytes used 10.23GiB
devid  1 size 100.00GiB used 12.05GiB path /dev/md127p2
```

- # btrfs fi du -s /

```
Total Exclusive Set shared Filename
107.69GiB  1.55GiB  7.83GiB /
```

- # df -h

```
/dev/md127p2  100G  11G  89G  11%
```


btrfs check

- Cannot check a btrfs mounted file system!
 - Can be forced, but if kernel changes file system, could be problems!
 - Reboot into “Rescue System”
- Identify partition
 - `btrfs check <partition block device>`

Boot from readonly snapshot

- Identify target snapshot with snapper
- Reboot
- Select that snapshot, boot read only
- Everything OK?
- snapper rollback