A practical ZFS setup

Replacing RAID 5 at home

How we store (large amounts of) data

- Stacked ‘vinyl record players’
- In tiny metal cases
- Playing bit compositions on metal ‘records’
- Using electromagnetism to write and read
- With faulty firmware and
- Flaky power

- Phantom writes (writing to the wrong place)
- Bit rot (random bit flips)
- Write-Hole (partly writing new data on RAID 5)

What I’m comparing and why

The Champion:
State of the Art thing to do

- RAID 5 in software or hardware
- Some ‘inferior’ file system on top
- Prebuilt or custom Linux thingy

The Underdog:
New kid on the block

- Striped Mirrors (RAID 10)
- Using ZFS
- Probably running FreeNAS

The XOR operation: Putting the ‘R’ into ‘RAID’

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

How this affects your NAS

- Redundancy relies on graceful failure:
  - Your file system trusts your RAID card to deliver the right data or an error
  - Your RAID card trusts its disks to return the right data or an error
  - Your disks have firmware to protect against basic errors
- A second disk dies when you rebuild your RAID 5
- No good upgrade story
- High initial cost
- No data integrity checks
- No transactional semantics

What does ZFS change

- Checksums for data and metadata (does not trust the disk)
- Always uses Copy-On-Write (never overwrites data in-place)
- Provides very cheap snapshots (zfs snap)
- FS level replication and backups (zfs send and zfs recv)
- Configurable compression and deduplication
- Keeps going in degraded state

- How is it not terribly slow: It eats all your RAM
  - 1 GB RAM per 1 TB disk without deduplication
  - 5 GB RAM per 1 TB disk with deduplication

Downsides

- Needs loads of RAM
- ECC would be nice
- Less hardware compatibility (On top of FreeBSD)
- Gets slow when your disk gets really full

The proposed setup

- FreeNAS
- Don’t change FreeNAS defaults
  - Do not enable deduplication
  - Consider not encrypting underlying disks
- Start with a single disk
- Add a mirror of that disk (same size)
- Add another pair of mirrors (arbitrary but same size)
- Until you run out of SATA ports or case real estate
- Then replace the smallest pair of disks

Demo

Resources

https://github.com/problame/talkintrozfs2016

https://app.media.ccc.de/v/gpn16-7633-an_introduction_to_zfs

http://www.jupiterbroadcasting.com/show/bsdnow/

http://mschwaig.github.io/