Snapshot Backups

Adam Thompson, Manitoba UNIX User Group, 2016-Feb-09

Remember - backward-looking only!

- Snapshots identify a moment in time for the affected system or filesystem
- The working copy remains the "master" copy, logically, and is read/write
- The snapshot is typically read-only.
- Snapshots can sometimes be read/write, too (this gets really confusing).
- The "current" state of the device/filesystem is usually stored as a blockwise diff against the most recent snapshot. Reading a block from disk *may* require traversing every snapshot ever created.

Filesystem Snapshot Technologies

- VM-level (typically QCOW2, CEPH, etc.)
- LVM
- ZFS
- Btrfs
- Hammer FS
- Proprietary

VM-level snapshots

- Easy (usually trivial) to perform
- Somewhat "proprietary" even when using open standards
  - difficult/impossible to manipulate/access outside hypervisor
- Simple operation, simple properties
- Demo (hopefully)
- Commands (for QCOW2):
  - Create: "qemu-img create -f qcow2 -b *originalfile*.img *snapshot*.img"
    - Note: the rare exception to the "backward-looking" property!
  - Remove: "rm *snapshot*.img"
  - Rollback: just shut down the VM, restart with *originalfile*.img.

Filesystem snapshots

- Mostly work pretty much the same way:
  - An epoch is declared/marked in the filesystem.
  - All new writes to the filesystem are stored as diffs against the last snapshot.
  - Older snapshot(s) remain accessible in the filesystem through magical directory names.

LVM snapshots

- Mature.
- Reliable.
- So incredibly badly documented it makes me want to cry.
- Demo (probably not)
- Cool application:
  - https://wiki.archlinux.org/index.php/Create_root_filesystem_snapshots_with_LVM
- Commands:
  - Create: lvcreate.
  - Remove: lvremove.
  - Rollback: lvconvert --merge (WTF?).

ZFS snapshots

- Mature.
- Reliable.
- Solaris (and now FreeBSD) actually *boot* from root filesystem snapshots.
  - "Boot Environments"
- ZOL is production-ready, not sure about booting though...
- Demo (maybe)
- Commands:
  - Create: "zfs snap something@*snapname*".
  - Remove: "zfs destroy *snapname*".
  - Rollback: "zfs rollback *snapname*".

Btrfs, HammerFS, etc.

- All work pretty much the same way, at least to the user.
- Maturity and reliability vary - neither is yet mainstream, so YMMV.
- But HAMMER does something totally different under the covers - not going to explain what/how/why. Active research project.
- No demo (definitely no demo).

Recover / Restore

- Usually trivial: one command to "roll back to snapshot XYZ"
- Demo (hopefully)