

OpenZFS on Linux Development

OpenSFS Lustre Developer Meeting

Jan 22th, 2015

Brian Behlendorf
behlendorf1@llnl.gov



LLNL-PRES-663196

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC



OpenZFS on Linux

- Current version 0.6.3 (released June 12th 2014)
- Easy to install packages for many distributions.



debian



fedora



gentoo linux™



ubuntu

- Large enthusiastic user community.
 - zfs-discuss@zfsonlinux.org
 - #zfsonlinux on freenode.net
 - <http://zfsonlinux.org>

OpenZFS on Linux – Version 0.6.3

- Near feature parity with other OpenZFS implementations.
- Systematically addressing gaps in functionality
- Wide spectrum of users
- Used on diverse hardware
- Contributions (0.6.2-0.6.3)
 - 58 different developers
 - 301 commits

Highlights

Updated Write Throttle

ARC Performance Improvements

POSIX ACLs

File Attributes (immutable, append-only)

Relatime style updates

SELinux Integration

Systemd Integration

ZFS Event Daemon (ZED)

Aarch64 and Sparc64 Support

Over 200 Bug fixes

OpenZFS on Linux – Version 0.6.4

- Continue to integrate OpenZFS features
- Continue to address known gaps
- Continue to improve Linux integration
- Contributions (0.6.3-HEAD)
 - 37 different developers
 - 141 commits

Planned Highlights

Feature Flag: Spacemap Histograms

Feature Flag: ZFS Bookmarks

Feature Flag: Hole Birth

Feature Flag: Embedded Data

Metaslab Improvements

Xattr Improvements

AIO and DirectIO Support

Fallocate Hole Punching

Linux Tracepoints

NFS access to .zfs/snapshot

100 bug fixes and counting

The Road to Version 1.0.0

- Minor releases have a development focus
 - 0.6.x – Functionality / Integration
 - 0.7.x – Memory Management
 - 0.8.x – ZFS+SPL Consolidation
 - 0.9.x – Stable ABI / Hardening
 - 1.0.x – Feature Development / Performance
- Longer term roadmap to guide development
- Current development activities continue in parallel
- Features and bug fixes are merged when ready

OpenZFS on Linux - Version 0.7.x

- Focus: Memory Management
- Goal: ARC / page cache integration
 - Data buffers will be backed by page vectors
 - Data pages will be mapped in to the page cache
- Benefits:
 - Uses standard Linux memory accounting mechanisms
 - Uses standard Linux memory reclaim mechanisms
 - Eliminates fragmentation overhead
 - Eliminates mmap double caching
 - 32-Bit platform support (x86, ARM)

OpenZFS on Linux - Version 0.8.x

- Focus: ZFS+SPL Consolidation
- Goal: Merging ZFS+SPL git repositories
 - Existing layering is preserved
 - An opportunity to define the ZFS kernel ABI
- Benefits:
 - Simpler packaging for users and maintainers
 - Eliminates the kmod dependency problem
 - Eliminates the risk of mismatched versions
 - One source tree for developers
 - Improves portability for non-Linux platforms

OpenZFS on Linux - Version 0.9.x

- Focus: Stable ABI / Hardening
- Goal: Finalize a stable user/kernel ABI
- Benefits:
 - Smoother upgrades / downgrades
 - Establishes a clear a user/kernel ABI for Linux
- Goal: Hardening
 - Gracefully handle a wider range of potential failure modes
 - Fault management via the ZFS Event Daemon (ZED)
- Benefits:
 - Enables deployment of lower end commodity hardware
 - Even more robust operation

OpenZFS on Linux - Version 1.0.x

- Focus: Feature development / performance
- Goal: Mature high quality code base
- Benefits:
 - Semantic Versioning
 - New feature development
 - Performance analysis



Development Model

- Project hosted at Github
 - <https://github.com/zfsonlinux/>
 - 1201 Watchers, 300 Forks
- illumos is tracked as upstream
- Independent of the Linux Kernel
 - Decouples ZFS from kernel updates
 - Linux 2.6.32 – 3.17 kernels supported
 - Enables use on non-Linux platforms
 - ZFS utilities / kmod can share code
 - Easier to integrate OpenZFS changes from Illumos/FreeBSD/OSX/OSV



Development Model – Issue Tracker

- Github issue tracker
 - Feature requests, bug reports, and milestones
 - Developers actively participate on the tracker
 - 539 open issues including 115 feature requests
 - Everything is as open and public as possible
 - Discussion by users and developers is encouraged
 - Issues are cross-linked to relevant git commits

Development Model - Milestones

🚩 6 Open 🚩 15 Closed		Sort ▼
0.6.4 No due date ⌚ Last updated 27 minutes ago ZFS on Linux v0.6.4 - Functionality / Integration	<div><div></div></div> 74% complete 101 open 284 closed Edit Close Delete	
0.7.0 No due date ⌚ Last updated 3 minutes ago ZFS on Linux v0.7.0 - Memory management	<div><div></div></div> 51% complete 90 open 95 closed Edit Close Delete	
0.8.0 No due date ⌚ Last updated 29 minutes ago ZFS on Linux v0.8.0 - ZFS+SPL Consolidation	<div><div></div></div> 32% complete 17 open 8 closed Edit Close Delete	
0.9.0 No due date ⌚ Last updated 1 minute ago ZFS on Linux v0.9.0 - Stable ABI / Hardening	<div><div></div></div> 17% complete 5 open 1 closed Edit Close Delete	
1.0.0 No due date ⌚ Last updated less than a minute ago ZFS on Linux v1.0.0 - Feature Development / Performance	<div><div></div></div> 0% complete 0 open 0 closed Edit Close Delete	

Development Model – Pull Requests

- Github pull requests
 - Used to submit proposed code changes
 - All proposed changes must be reviewed
 - Continuous integration development model
 - Proposed changes are automatically tested using buildbot
 - Developers get quick feedback on any proposed change
 - Good test coverage (kernel, architecture, distribution, etc)
 - Changes are tested a second time after being merged
 - The master branch is ***always*** kept stable
 - Designed to make it easy for anyone to contribute

Development Model – Pull Requests

Linux tracepoint integration #2874

Edit

New issue



nedbass wants to merge 4 commits into `zfsonlinux:master` from `nedbass:b_tracepoints`

Conversation 0

Commits 4

Files changed 18

+1,229 -137



nedbass commented a day ago

Collaborator

See #2406

Changes from the PR by @prakashsurya:

- Implement `dprintf()` as a tracepoint.
- Implement two new `DTRACE_PROBE`'s
- Refactor patch stack and make it `cstyle`-compliant
- Address feedback from @behlendorf on #2406

One thing I'm concerned about is the 256 byte stack buffer used by `dprintf()`. While short-lived this could potentially push stack usage over the limit. I've seen messages as long as 202 bytes to this isn't an unreasonable size, but we may want to consider dynamic allocation if this turns out to be problematic.



nedbass and others added some commits 17 days ago

- Move a few internal ARC structures to `arc_impl.h` ...
- Fix `dprintf` format specifiers ...
- `cstyle`: allow right paren on its own line ...
- Swap `DTRACE_PROBE*` with Linux tracepoints ...

• f772ef3
• 4d0e3fc
• 1be665c
• 7bde2d2

Labels

Difficulty - Medium

Feature

Milestone

0.6.4

Assignee

No one—assign yourself

Notifications

Unsubscribe

You're receiving notifications because you were mentioned.

3 participants



Console View

Legend: Passed Failed Warnings Failed Again Running Exception Offline No data

Go

15
LLNL-PRES-663196

Join Us, Contributors Welcome

- If you are a developer...
 - Port a change from Illumos/FreeBSD/OSX/OSV
 - Review or comment on a proposed pull requests
 - Implement a requested feature or fix a known issue
 - Help us improve the automated testing
- If you are a user...
 - Open a new issue if you encounter a problem
 - Open pull requests even for trivial fixes
 - Help us rigorously test new features and bug fixes

Questions

