Optimizing Teuthology:

Turning failures into faster Ceph fixes

ljsanders@uk.ibm.com

The Problem

Teuthology jobs where IO is being injected into ceph eg. FIO, rados bench, ceph_test_rados

If an OSD assert occurs, and the client has hung as a result. The client is left hanging because the watchdog has not detected the OSD has asserted

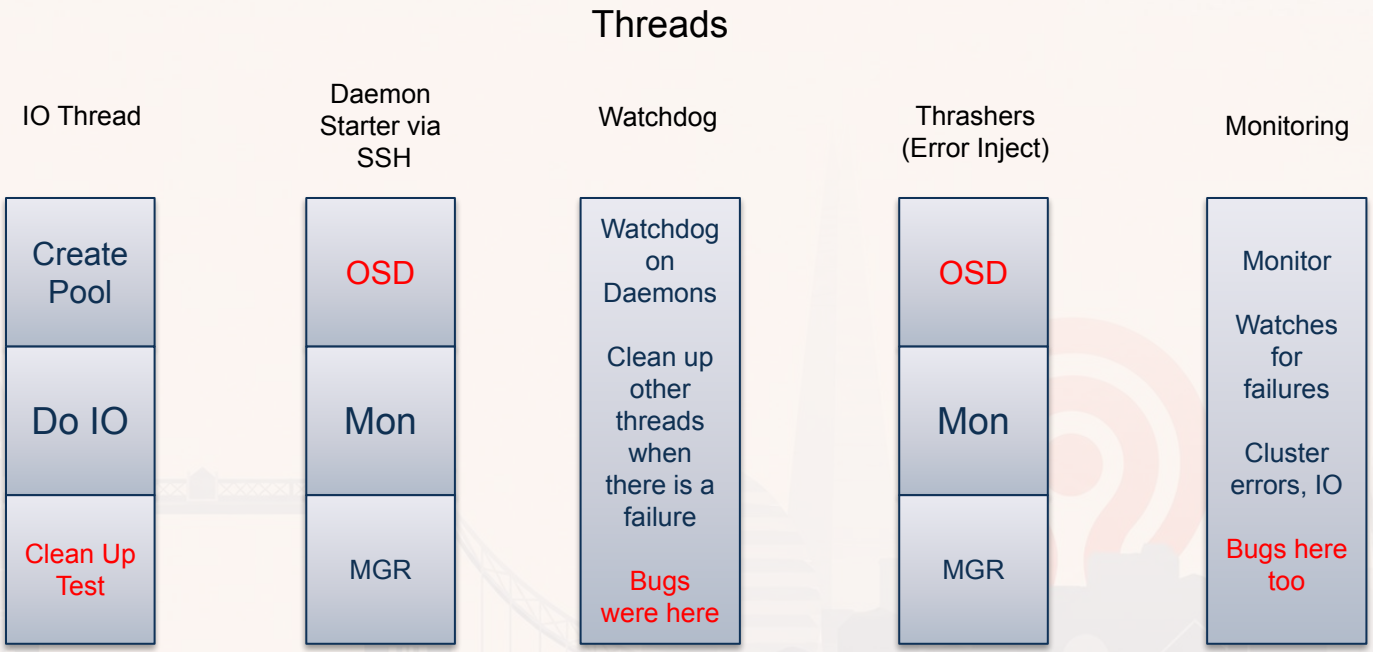
This results in a dead job for 8+ hours

The system allocated to running the teuthology job is locked until teuthology kills the job

The developer isn't notified of the failure until **all** jobs in the teuthology-suite run have either Passed, failed or been marked as dead

This is a waste of developer time and test lab resources

Teuthology Job Layout



The Fixes

There is a watcher that looks for errors from the thrasher and cluster logs which shutdown the IO client

Update watcher to react better if an OSD asserts, it notified the test to kill the IO clients

Improved test clean up, to not attempt to run clean up processes like delete the pool if the test has failed. Move onto deleting the cluster

Better failure reasons

The failure reason for the test now shows which thrasher/process/Ceph daemon failure that caused the watchdog to fire, instead of a python backtrace of whatever process failed when the test was torn down

Assists debug of what went wrong instead of a Dead job. Converts dead into fail at the point the failure occurred and the system can be reallocated instead of hung for 8+ hours

A bad build with an OSD assert doesn't consume a lot of test labs resources and developer gets their test result a lot faster, resulting in improved developer efficiency to write more code

Summary

- https://github.com/ceph/ceph/pull/64889

- Spotting a test failure much earlier means the system can be used for next job in the queue. More efficient system utilization
- Number of dead jobs is much lower. Dead jobs lock a machine up for 8 hours
- Developers get their test results a lot quicker – faster delivery of PRs
- Screening of test failures consume a lot of developer time. Highlighting the failure reason reduces the amount of time to screen the failure and screening of duplicates

Thank you!!!!

ljsanders@uk.ibm.com

Slack: @ljsanders