autoboot

Sep 2006

Andi Kleen, SUSE Labs
ak@suse.de

Motivation

- Regular regression testing of Linux kernel
- Regular snapshots of mainline x86-64 tree
  - Pulled into -mm* and used by a lot of people
  - Need to do more simple pre-testing and regular tests of larger testsuites
  - Test matrix increases with more i386 work
- Use more build power for compiling of different configs
  - Just make sure everything still builds
  - CONFIG_* spagetti in the kernel is a big maintenance issue
  - But can throw CPU power and scripts at it
    - Had that before but didn’t do it always because it took too long
- Earlier regression test results
- Also wanted history of regression test/performance
- http://test.kernel.org good start
    - But doesn’t cover enough machines or configurations
- Got tired of waiting

autoboot

- Building/Booting/testing is actually quite mechanic
  - Build static kernel, upload, grubonce, reboot, wait, test, wait
  - At least as long everything goes right...
  - If anything goes wrong power switch, grab logs
- autoboot is a shell script collection to automate all this
  - Coupled with paranoid error handling
- It basically does what I did by hand before
  - So I know the procedure works
- Primary target mainline/x86-64 kernels
  - After all near all changes happen in mainline
    - Low frequency and few big merge points where it is too late
    - Still could be used for it with changes
  - And regressions are the easier to fix the sooner they are detected

Implementation

- Core
  - buildtestkernels
  - autoboot kernel [host] testscript
  - Various helpers and wrappers
  - Uses query_powerswitch* scripts from trenn
- Master builds trees and controls the tests
  - Configurations auto generated by tweaking defconfig
    - Generates unique identifiers for each kernel tree
  - Fetches logs using ssh from console server
  - Controls clients by ssh and manages timeouts
  - Runs under own user in the background
  - All logs for a session saved in unique directory for kernel/configuration
- Clients
  - Need power switch/console/ssh key/grub entry

Client side

- Installs/updates all the other software itself in /usr/local
  - Except for the scripts which are in /suse right now
- Mostly using autotest (http://test.kernel.org/autotest)
  - Python framework
  - Runs various benchmarks and stress tests with error checking
  - LTP, lmbench, dbench, ...
- Also have a couple of own scripts
  - Want more: reaim, ballista, LTP quickhit, stress2
  - Still looking for a good <=1 minute stress test
- Easy to add new client scripts
  - And easy to wrap existing tests in this

Problems

- Sometimes autotest hang problem
- Generally stresses test machines/infrastructure more
  - Run into various hardware failures
  - Currently some machines dead
- Need faster build system for master
  - icecream in large network seems to be slower for kernel builds
- In some corner cases better error handling
  - e.g. don’t powerswitch when ssh hostkey wrong
- Improve host scheduler
- Sometimes oldconfig still needs human input

- But overall it works quite nicely

Future

- Support 32bit testing
  - Likely with nfsroot
  - Unfortunately install into directory is broken in SLES10
- Do builds and testing in parallel
- Implement automatic git bisect/quilt binary search for bugs on top of autoboot
  - Just needs a script to reproduce the problem
  - And finds out which patch added the bug
  - Should be fairly easy now
- HTML frontend
- Plot the performance data generated by autotest
  - Want to have long term performance trends

Summary

- Works well
  - Builds lots of different configurations for i386/x86-64
  - Boots four x86-64 configurations
- Code (mostly) fairly clean
  - And some documentation exists
- Cronjob running now that builds and boots mainline each night
- Manually triggered x86-64 tree builds
  - Plan to automate it with a staging tree
    - Goal <15min releases and larger regression tests nightly
- Questions?