Integrating ftrace into the perf tools

Namhyung Kim

2013-10-23 Wed

Who am I?

- Namhyung Kim
- working for LG Electronics
    - we have a small upstream team
- contributing to perf since 2012
    - just wanted to know how to use perf well :)

motivation

- on ELC this year
  - Steven Rostedt said about kbuffer in libtraceevent
  - he wanted other tools have access to ftrace
- perf and ftrace have commonalities
  - at least for the tools side
- so I started perf ftrace
  - in a hope of taking over trace-cmd ;-)

perf

- PMU counters
- tracepoint events
- software events, hardware breakpoints
- sampling based
- kernel + userspace
- perf tools live in kernel

ftrace

- function tracing
  - function tracer also exports an event
  - function graph tracer does not
  - other tracers: wakeup, preempt, irq, ...
- event tracing (tracepoint)
- can trace every occurrence
- kernel only
- trace-cmd/kernel-shark

libtraceevent

- library for parsing event tracing output
    - format exported on sysfs
    - factored out from trace-cmd
    - also used for perf, powertop, . . .
- kbuffer
    - ftrace output format parser

perf ftrace

- focus on the function graph tracer
  - don't make any use of perf events as of now
  - using kbuffer in libtraceevent
- use same command of perf
  - 'record' and 'report'
  - 'show' was added to see traditional output
  - 'live' requires no recording
- https://lkml.org/lkml/2013/10/16/15

issues

- multiple file storage
- perf event
- clock
- function filter
- event triggering/toggling
- multiple ftrace buffer

future works

- integrate with perf trace
- user-level function tracing ?
  - the binary compiled with -pg
  - trace_marker or SDT
- UI (like kernel-shark) ?
- and so on