Discover what your app is waiting with kernel tracing

Tracing Summit 2013

Francis Giraldeau
francis.giraldeau@gmail.com

DORSAL Lab, École Polytechnique de Montréal
apt-get install tree

Traditional profiler output

Running : 63%
Blocking : 37%

What the app is waiting for?
What is the server doing?
UNIVERSAL* DISTRIBUTED PROFILER

*Requires Linux
Task state

sys_entry  sys_exit
write

sys_entry
PREEMPT
sched_switch

sys_entry
read
sched_switch

sys_exit
BLOCKED

TASK_INTERRUPTIBLE
TASK_UNINTERRUPTIBLE
Device wake-up

Task wake-up
Remote wake-up

- read
- BLOCKED
- Softirq
- Network
- Write

Wakeup signal flows through the system.
Required events

- sched_switch
- sched_wakeup
- irq_handler_{entry, exit}
- hrtimer_expire_{entry, exit}
- softirq_{entry, exit}
- inet_sock_local_{in, out}

System calls not required!
Analysis overview

Distributed tracing
1) Start relayd
2) Start tracing on each machine
3) Execute the load
4) Stop tracing

Off-line analysis in TMF
1) Create experiment
2) Synchronize traces
3) Compute the execution graph
4) Compute the critical path
Demo
Conclusion

- Crossing machines boundaries works
- Trace synchronization appropriate
- Shed light into complex, actual executions
- Precise and universal tool, independent from user-space runtime
Future work

- Performance counters of the active path
- Filter lttng-relayd packets for streaming
- More scalable analysis using state history
Thanks to Professor Michel Dagenais and our partners EfficiOS and Ericsson.

Special thanks to Geneviève Bastien for her excellent work on Luna Dorsal.

More information on:

http://step.dorsal.polymtl.ca/~fgiraldeau
Annexes
RPC Server

Commands: hog

Default control flow view

Critical Flow View: request hog()

Critical Flow View: request hog()
1) GET /wkdb/polls/1
2) POST vote
3) GET redirect

question 1

- choice 2
- choice 1

Vote

question 1

- choice 2 -- 26 votes
- choice 1 -- 13 votes

Vote again?
What the server is doing?
Django Polls App
mechanize + Apache WSGI + postgresql

1. GET form
2. POST form
3. redirect
Django Polls App
mechanize + Apache WSGI + postgresql

1) GET form
Django Polls App
mechanize + Apache WSGI + postgresql

2) POST form
Django Polls App
mechanize + Apache WSGI + postgresql

3) GET redirect
Network share

Mount CIFS ~330ms

- modprobe
- Samba server
- kworker
- kthreadd
- jbd2/sda1-8
- Samba server
Network share
directory listing

ls sends packets from newstats() and getents()