LTTng

The *Linux Trace Toolkit: next generation* is an open source software toolkit which you can use to simultaneously trace the Linux kernel, user applications, and user libraries.

http://lttng.org/features/
LTTng supports architectures such as IA-32 (x86), x86-64, PowerPC, ARM, and MIPS, among others. ...

The LTTng Documentation
http://lttng.org/docs/v2.10/#doc-tracing-your-own-user-application
Howto tracing with LTTng – tutorial covers installation and basic usage.

Kernel Trace with LTTng

Source: http://lttng.org/docs/v2.10/#doc-tracing-the-linux-kernel

Before you follow this guide, make sure to install LTTng. [including kernel modules for LTTng kernel tracing].
...

[Basic required pkgs:
  lttng-tools
  lttng-modules-dkms
]

The following command lines start with the # prompt because you need root privileges to trace the Linux kernel. You can also trace the kernel as a regular user if your Unix user is a member of the tracing group.

1. Create a tracing session which writes its traces to /tmp/my-kernel-trace:

```
# lttng create my-kernel-session --output=/tmp/my-kernel-trace
```

2. List the available kernel tracepoints and system calls:

```
# lttng list --kernel
# lttng list --kernel --syscall
```

3. Create event rules which match the desired instrumentation point names, for example the sched_switch and sched_process_fork tracepoints, and the **open**(2) and **close**(2) system calls:

```
# lttng enable-event --kernel sched_switch,sched_process_fork
# lttng enable-event --kernel --syscall open,close
```

You can also create an event rule which matches *all* the Linux kernel tracepoints (this will generate a lot of data when tracing):

```
# lttng enable-event --kernel --all
```

4. Start tracing:

```
# lttng start
```

5. Do some operation on your system for a few seconds. For example, load a website, or list the files of a directory.

6. Stop tracing and destroy the tracing session:

```
# lttng stop
# lttng destroy
```

The **lttng-destroy**(1) command does not destroy the trace data; it only destroys the state of the tracing session.

7. For the sake of this example, make the recorded trace accessible to the non-root users:

```
# chown -R $(whoami) /tmp/my-kernel-trace
```

See View and analyze the recorded events to view the recorded events.

Also See:

- How to use LTTng to diagnose problems
- http://archive.eclipse.org/tracecompass/doc/stable/org.eclipse.tracecompass.doc.user/LTTng-Kernel-Analysis.html
- LTTng built-in help system: http://127.0.0.1:42105/help/index.jsp

A lttng kernel trace session example

```
# lttng create k1-lttng --output=/tmp/k1-lttng.trc
# lttng list --kernel   << see all possible kernel events >>
Kernel events:
-------------
      scsi_dispatch_cmd_start (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      scsi_dispatch_cmd_error (loglevel: TRACE_EMERG (0)) (type: tracepoint)
```

```
...
      sched_pi_setprio (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      signal_generate (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      signal_deliver (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      skb_kfree (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      skb_consume (loglevel: TRACE_EMERG (0)) (type: tracepoint)
...
      block_rq_issue (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      block_bio_bounce (loglevel: TRACE_EMERG (0)) (type: tracepoint)
...
      gpio_value (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      irq_handler_entry (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      irq_handler_exit (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      irq_softirq_entry (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      irq_softirq_exit (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      irq_softirq_raise (loglevel: TRACE_EMERG (0)) (type: tracepoint)
...
      kmem_kmalloc (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      kmem_cache_alloc (loglevel: TRACE_EMERG (0)) (type: tracepoint)
...
      kvm_async_pf_completed (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      module_load (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      module_free (loglevel: TRACE_EMERG (0)) (type: tracepoint)
...
      writeback_wait_iff_congested (loglevel: TRACE_EMERG (0)) (type: tracepoint)
      writeback_single_inode (loglevel: TRACE_EMERG (0)) (type: tracepoint)
#

# lttng enable-event --kernel --all   << trace all! Big files.. >>
# lttng start ; sleep 1; lttng stop
```

Viewing the trace

```
# babeltrace /tmp/k1-lttng.trc/kernel/
[11:40:08.863740605] (+?.??????????) kaiwan-T460 kmem_kmalloc: { cpu_id = 0 },
{ call_site = 0xFFFFFFFFC10AA457, ptr = 0xFFFF930CC5098000, bytes_req = 739, bytes_alloc
= 1024, gfp_flags = 20971712 }
[11:40:08.863747488] (+0.000006883) kaiwan-T460 kmem_kfree: { cpu_id = 0 }, { call_site
= 0xFFFFFFFFC10AA52A, ptr = 0xFFFF930CC5098000 }
[11:40:08.863749701] (+0.000002213) kaiwan-T460 sched_waking: { cpu_id = 0 }, { comm =
"lttng-consumerd", tid = 32437, prio = 20, target_cpu = 3 }
[11:40:08.863752608] (+0.000002907) kaiwan-T460 sched_wakeup: { cpu_id = 0 }, { comm =
"lttng-consumerd", tid = 32437, prio = 20, target_cpu = 3 }
[11:40:08.863754831] (+0.000002223) kaiwan-T460 power_cpu_idle: { cpu_id = 3 }, { state
= 4294967295, cpu_id = 3 }
[11:40:08.863756344] (+0.000001513) kaiwan-T460 kmem_kmalloc: { cpu_id = 0 },
{ call_site = 0xFFFFFFFFC10AA457, ptr = 0xFFFF930C7FAEAA00, bytes_req = 275, bytes_alloc
= 512, gfp_flags = 20971712 }

...

[11:40:09.948720350] (+0.000001293) kaiwan-T460 power_cpu_idle: { cpu_id = 1 }, { state
= 1, cpu_id = 1 }
[11:40:09.948720370] (+0.000000020) kaiwan-T460 syscall_entry_fcntl: { cpu_id = 0 },
{ fd = 46, cmd = 2, arg = 1 }
[11:40:09.948721513] (+0.000001143) kaiwan-T460 syscall_exit_fcntl: { cpu_id = 0 },
{ ret = 0, arg = 1 }
[11:40:09.948722298] (+0.000000785) kaiwan-T460 syscall_entry_setsockopt: { cpu_id =
0 }, { fd = 46, level = 1, optname = 16, optval = 139758116252252, optlen = 4 }
[11:40:09.948723921] (+0.000001623) kaiwan-T460 syscall_exit_setsockopt: { cpu_id = 0 },
{ ret = 0 }
[11:40:09.948728342] (+0.000004421) kaiwan-T460 syscall_entry_recvmsg: { cpu_id = 0 }, {
fd = 46, msg = 139758116252128, flags = 0 }
```

. . .

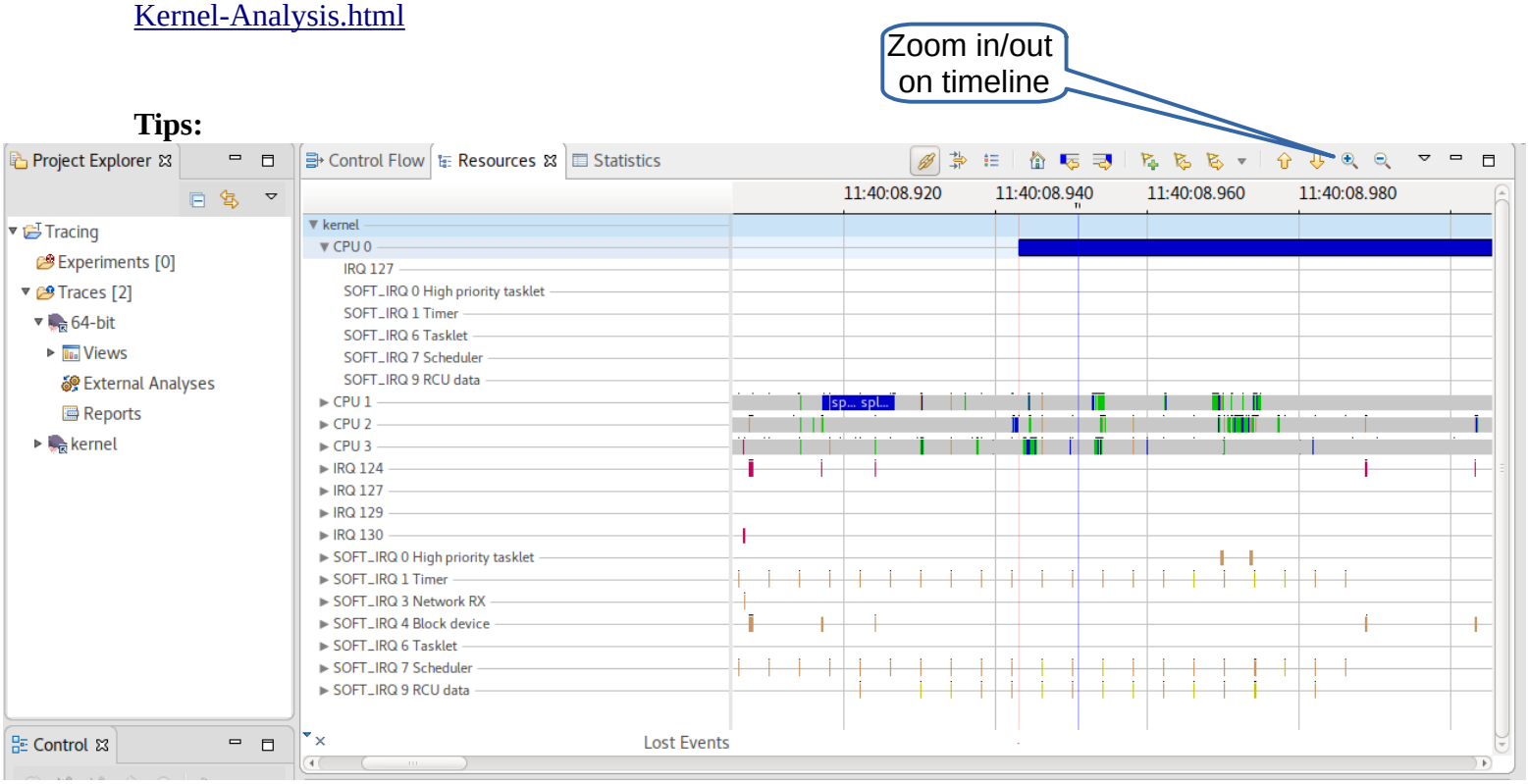
#

It’s all there, just not very human readable…
So we use the ***TraceCompass*** ***GUI***. Much, much better!

Using the TraceCompass GUI!

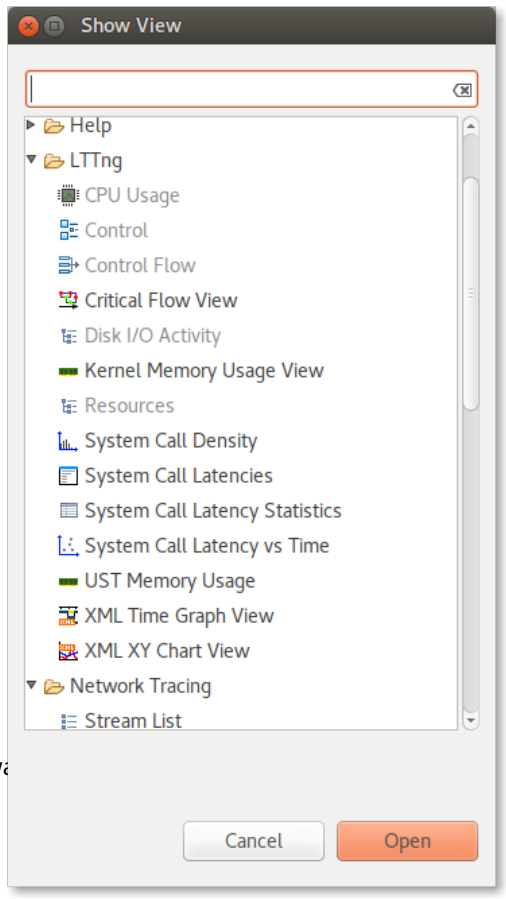
Doc: https://archive.eclipse.org/tracecompass/doc/stable/org.eclipse.tracecompass.doc.user/LTTng-Kernel-Analysis.html

Tips:

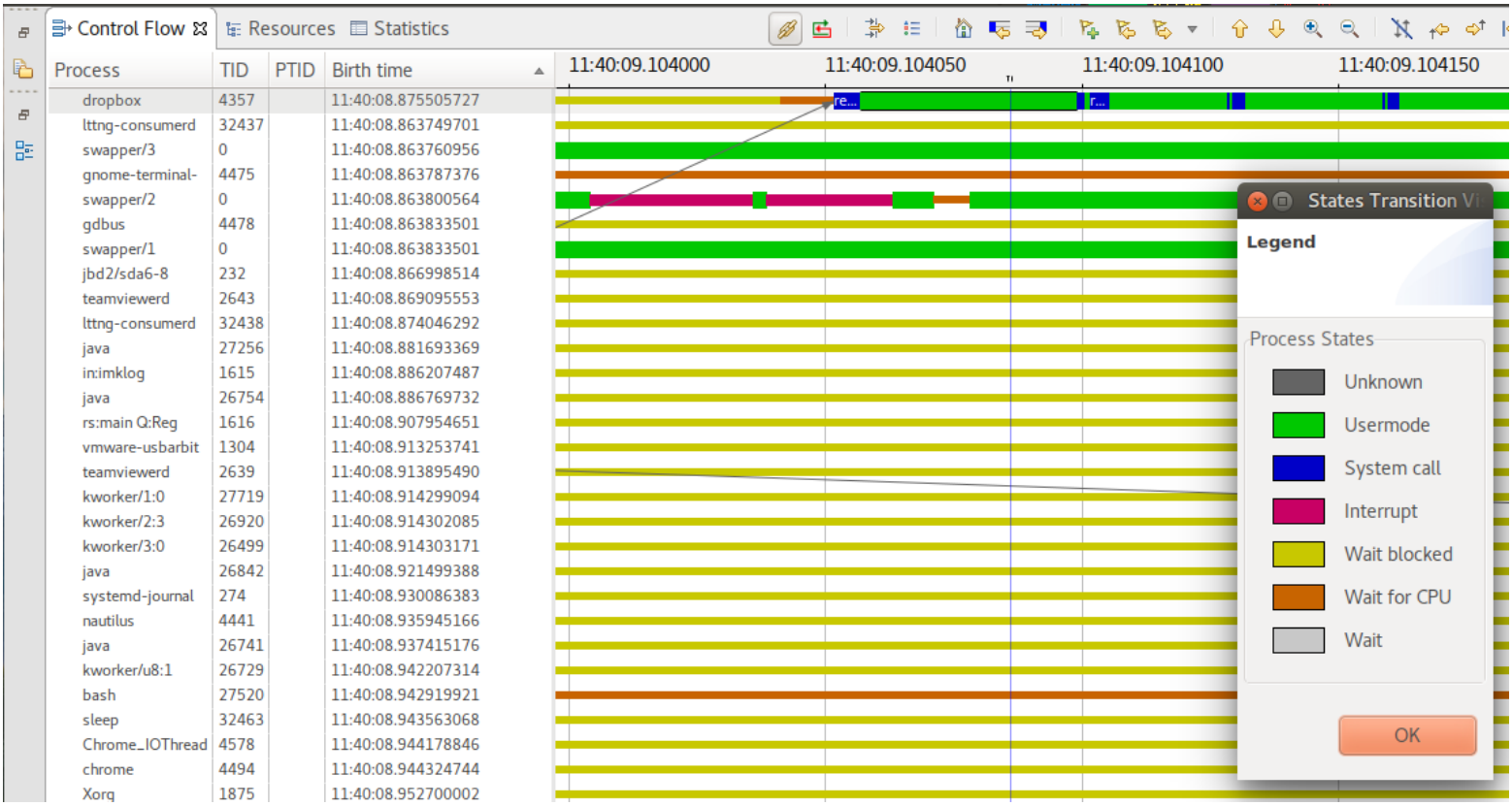


Window / Show Perspective / LTTng Kernel (default)

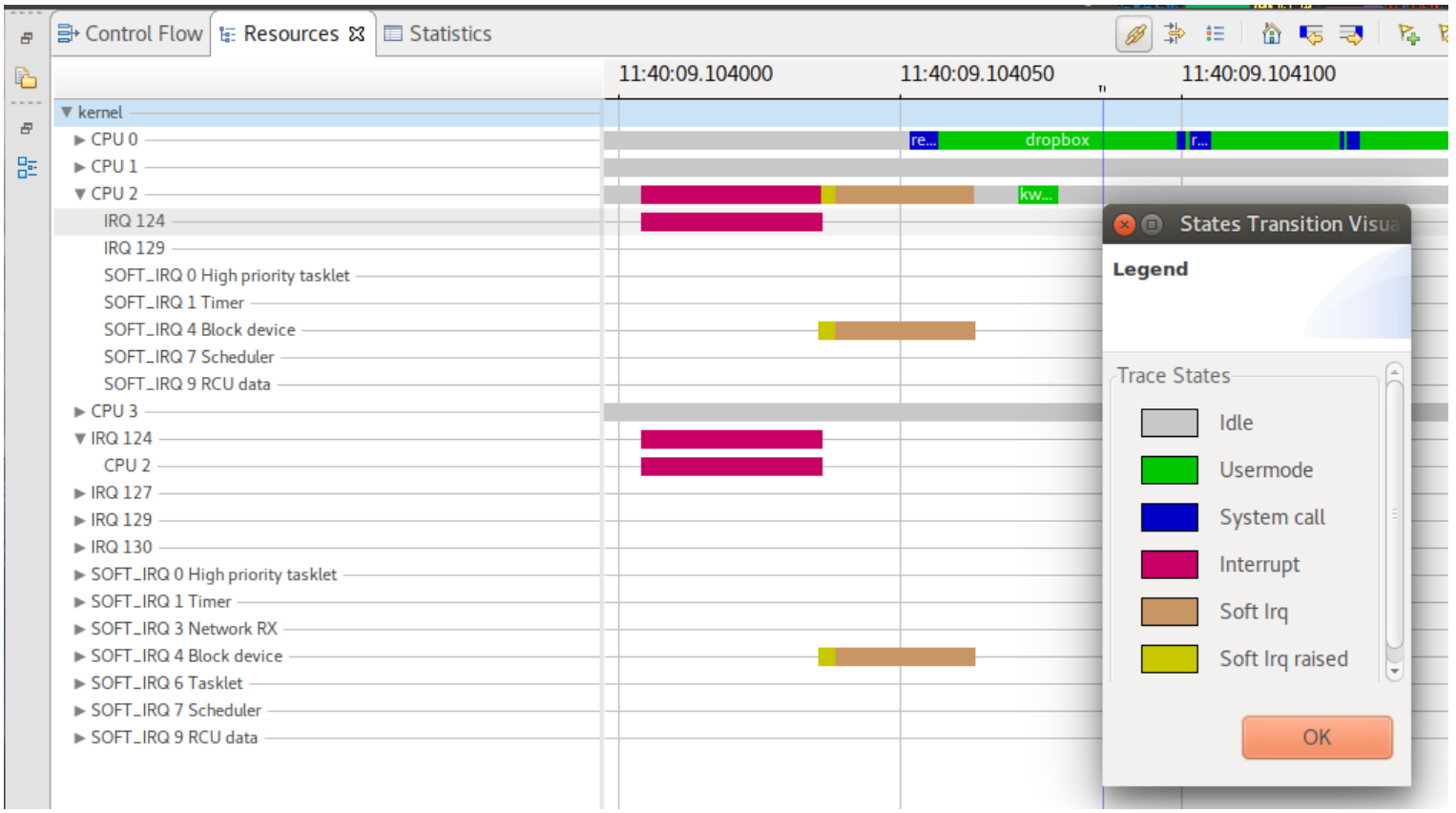
Window / Show View :



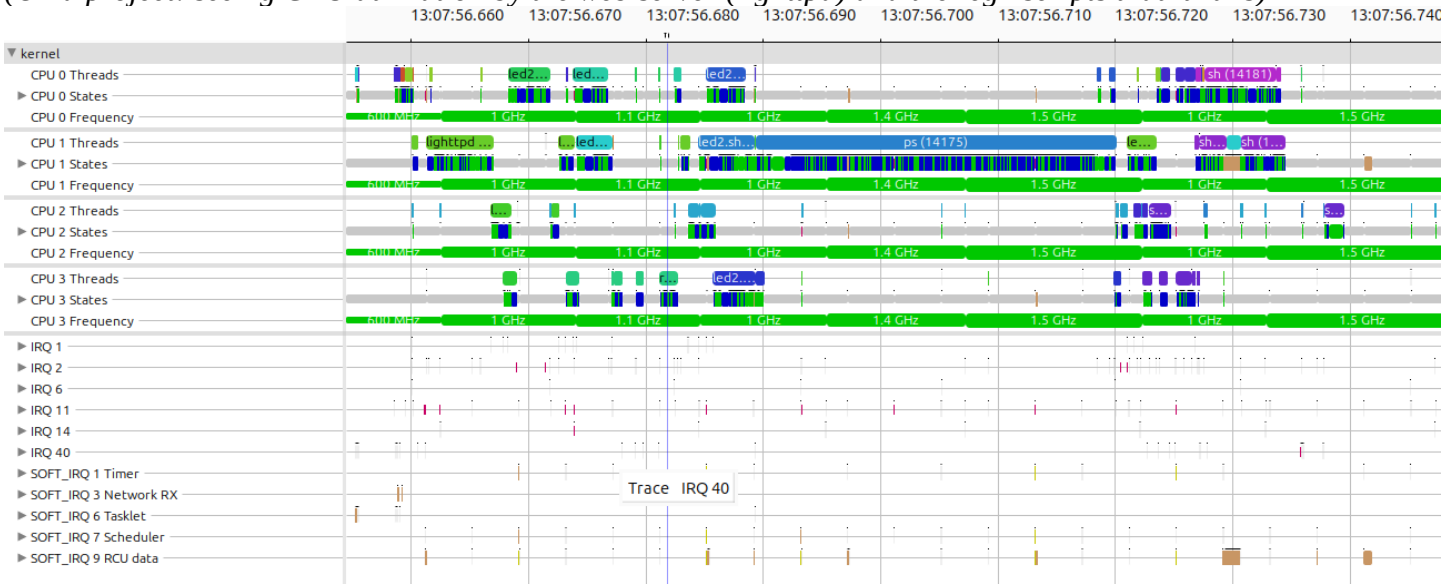
'Control Flow' tab



'Resources' tab

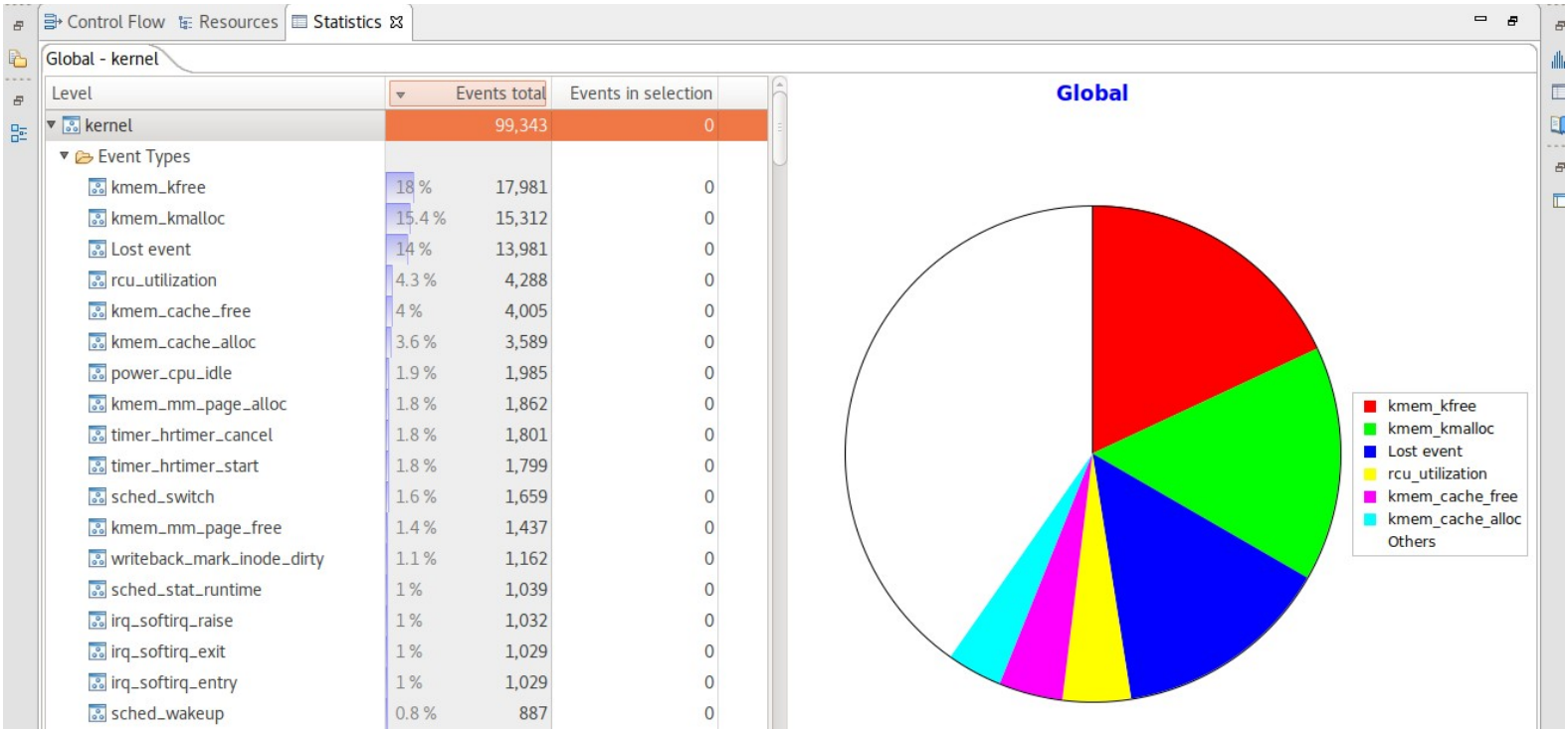


(Tip: in the first screenshot, an arrow can be seen; it represents a scheduling switch. From the docs: *"… The arrows indicate when the scheduler switches from one process to another for a given CPU…"*).
(On a project: seeing CPU utilization by the web server (lighttpd) and the 'cgi' scripts that it runs)

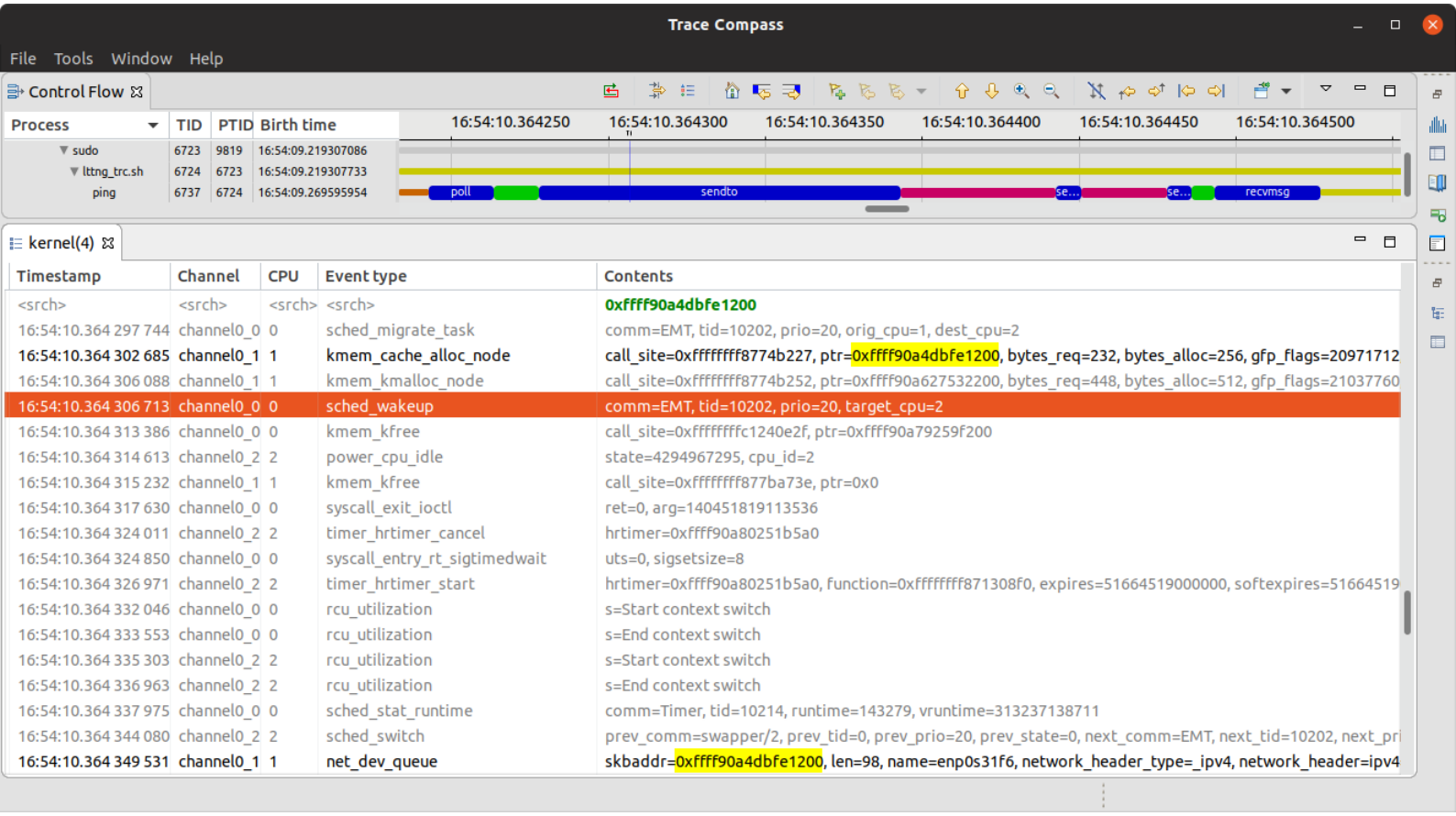


(We can perhaps think: why's 'ps' taking an inordinately long time here?)

'Statistics' tab



Below: we're following the *ping* process; we have found it issuing the sendto() syscall. Within it, spotted the kmem_cache_alloc_node(); the return pointer is seen here (obtained by rt-click on the line / *Copy to Clipboard*):

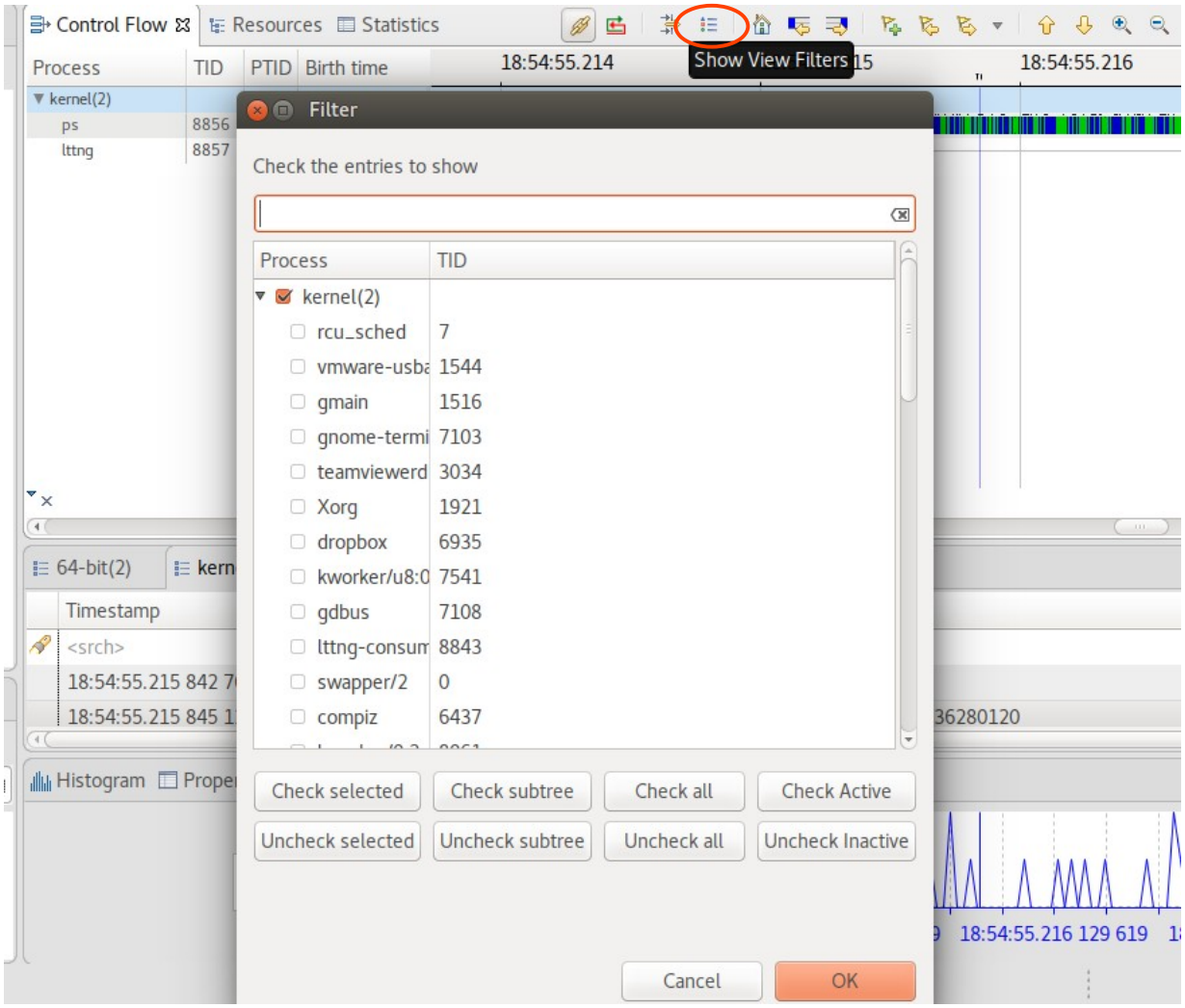



```
Timestamp       Channel CPU   Event type  Contents    TID   Prio  PID
16:54:10.364 302 685           channel0_1   1   kmem_cache_alloc_node
      call_site=0xffffffff8774b227, ptr=0xffff90a4dbfe1200, bytes_req=232,
bytes_alloc=256, gfp_flags=20971712, node=-1   6737  20    6737
```

The ptr value is filtered upon (above screenshot); we can now literally *see* for ourselves where it's used next (its highlighted); in net_dev_queue(); following it down further, one comes to the free call site (the kmem_cache_free() on this same ptr)!

Also, the parameters are displayed ! Note that they aren't necessarily parameters, but very very useful information!

Also:
Click the "Show View Filters" button; a dialog pops up allowing one to filter entries. Select entries of interest and check them out! *(in the screenshot below, among processes alive at the time of the trace, we selected only 'ps' and 'lttng')*:


Control Flow
Resources
Statistics
Show View Filters
Process
TID
PTID
Birth time
18:54:55.214
18:54:55.216
kernel(2)
ps 8856
lttng 8857
Filter
Check the entries to show
Process
TID
kernel(2)
rcu_sched 7
vmware-usba 1544
gmain 1516
gnome-termi 7103
teamviewerd 3034
Xorg 1921
dropbox 6935
kworker/u8:0 7541
gdbus 7108
lttng-consum 8843
swapper/2 0
compiz 6437
Check selected
Check subtree
Check all
Check Active
Uncheck selected
Uncheck subtree
Uncheck all
Uncheck Inactive
Cancel
OK
64-bit(2)
Timestamp
<srch>
18:54:55.215 842 7
18:54:55.215 845 1
Histogram
18:54:55.216 129 619

Additional CLI Profiling Tools

Very powerful CLI tools available too!
Check this out:

```
$ lttng<tab><tab>
lttng                     lttng-iolatencytop       lttng-irqlog-mi
lttng-periodstats         lttng-schedstats
lttng-analyses-record     lttng-iolatencytop-mi    lttng-irqstats
lttng-periodstats-mi      lttng-schedstats-mi
lttng-cputop              lttng-iolog              lttng-irqstats-mi
lttng-periodtop           lttng-schedtop
lttng-cputop-mi           lttng-iolog-mi           lttng-memtop
lttng-periodtop-mi        lttng-schedtop-mi
lttng-crash               lttng-iousagetop         lttng-memtop-mi
lttng-relayd              lttng-sessiond
lttng-iolatencyfreq       lttng-iousagetop-mi      lttng-periodfreq
lttng-schedfreq           lttng-syscallstats
lttng-iolatencyfreq-mi    lttng-irqfreq            lttng-periodfreq-mi
lttng-schedfreq-mi        lttng-syscallstats-mi
lttng-iolatencystats      lttng-irqfreq-mi         lttng-periodlog
lttng-schedlog            lttng-track-process
lttng-iolatencystats-mi   lttng-irqlog             lttng-periodlog-mi
lttng-schedlog-mi
   << mi = machine interface >>
$ lttng^C
```

```
$ lttng-cputop /tmp/k1-lttng.trc/
Checking the trace for lost events...
[warning] Tracer discarded 12989 events between [11:40:08.873895341] and
[11:40:08.943080657] in trace UUID 72e8b8b633cc44f90422b6f534500, at path: "/tmp/k1-
lttng.trc/kernel", within stream id 0, at relative path: "channel0_0". You should
consider recording a new trace with larger buffers or with fewer events enabled.
[warning] Tracer discarded 992 events between [11:40:08.943080657] and
[11:40:09.485410272] in trace UUID 72e8b8b633cc44f90422b6f534500, at path: "/tmp/k1-
lttng.trc/kernel", within stream id 0, at relative path: "channel0_0". You should
consider recording a new trace with larger buffers or with fewer events enabled.
Processing the trace: 100%
[#########################################################################################
##############] Time: 0:00:01
Timerange: [2017-08-25 11:40:08.863740605, 2017-08-25 11:40:09.941728585]
Per-TID Usage
Process                    Migrations   Priorities
###############################################################################
                                                                                4.81 %
lttng-consumerd (32438)             1   [20]
                                                                                4.13 %
indicator-multi (3794)              2   [20]
                                                                                1.80 %
(3707)                              5   []
                                                                                1.77 %
dropbox (4357)                      5   [20]
                                                                                1.29 %
kworker/u8:1 (26729)                2   [20]
                                                                                0.81 %
Xorg (1875)                         0   [20]
```

```
██████                                                                0.46 %
lttng-consumerd (32437)          2   [20]
████                                                                  0.26 %
teamviewerd (2632)               5   [20]
███                                                                   0.22 %
chrome (4494)                    3   [20]
███                                                                   0.20 %
unity-panel-ser (3715)           1   [20]

Per-CPU Usage
################################################################################
████████████████████████████████████████████████████████████████████████████ 24.62 %
CPU 0
██████████                                                             4.04 %
CPU 1
████████████████                                                       6.44 %
CPU 2
███████████████                                                        6.01 %
CPU 3


Total CPU Usage: 10.28%
$
$ lttng-memtop /tmp/k1-lttng.trc/
Checking the trace for lost events...
[warning] Tracer discarded 12989 events between [11:40:08.873895341] and
[11:40:08.943080657] in trace UUID 72e8b8b633cc44f90422b6f534500, at path: "/tmp/k1-
lttng.trc/kernel", within stream id 0, at relative path: "channel0_0". You should
consider recording a new trace with larger buffers or with fewer events enabled.
[warning] Tracer discarded 992 events between [11:40:08.943080657] and
[11:40:09.485410272] in trace UUID 72e8b8b633cc44f90422b6f534500, at path: "/tmp/k1-
lttng.trc/kernel", within stream id 0, at relative path: "channel0_0". You should
consider recording a new trace with larger buffers or with fewer events enabled.
Processing the trace: 100%
[##############################################################################
##############] Time: 0:00:01
Timerange: [2017-08-25 11:40:08.863740605, 2017-08-25 11:40:09.941728585]
Per-TID Memory Allocations
Process
################################################################################
████████████████████████████████████████████████████████████████████████████ 1024
pages lttng-consumerd (32438)
████████████████                                                       233
pages dropbox (4357)
███                                                                     50
pages bash (32463)
██                                                                      25
pages Xorg (1875)
██                                                                      21
pages indicator-multi (3794)
██                                                                      18
pages lttng-consumerd (32437)
                                                                        9
pages in:imklog (1615)
                                                                        8
pages rs:main Q:Reg (1616)
                                                                        3
pages lttng (32462)
                                                                        3
pages NetworkManager (1404)

Per-TID Memory Deallocations
Process
################################################################################
```

```
███████████████████████████████████████████████████████████████████████████████  960
pages lttng-consumerd (32438)
████████████                                                                     146
pages lttng (32462)
████                                                                              54
pages bash (32463)
█                                                                                 21
pages indicator-multi (3794)
█                                                                                 15
pages lttng-consumerd (32437)
                                                                                   4
pages kworker/2:3 (26920)
                                                                                   3
pages NetworkManager (1404)
                                                                                   2
pages jbd2/sda6-8 (232)
                                                                                   2
pages wpa_supplicant (1977)
                                                                                   2
pages gdbus (3258)

Total memory usage:
- 1405 pages allocated
- 1214 pages freed
$
```

Similarly, try out the lttng-iolatencytop / lttng-iousagetop / lttng-schedtop / lttng-periodtop tools.

Userspace Tracing

Requires adding instrumentation and tracepoints into the project codebase.
UST = User Space Tracer

***lttng-ust** — Linux Trace Toolkit Next Generation User-Space Tracer 2.x*
