



Contribution ID: 18

Type: **not specified**

## CPU Isolation MC

CPU Isolation can be considered as an holistic functionality that stems from a close combination of several kernel and userspace components working together to shield workloads with extreme latency or performance requirements from interruptions (also known as Operating System noise). An example of such type of workloads are DPDK (Data Plane Development Kit) use cases (Telco/5G) where even the shortest interruption (e.g., a few microseconds to service an IPI) can cause packet losses, eventually leading to exceeding QoS requirements.

Despite considerable improvements in the last few years towards implementing full CPU Isolation (nohz\_full, rcu\_nocb, isolcpus, etc.), a certain amount of issues remain to be addressed, as it is still relatively simple to highlight sources of OS noise just by running synthetic workloads mimicking polling (always running) type of application similar to the ones mentioned above.

Recently, Improvements and discussions about CPU isolation features have been made and discussed in LKML [1,2,3], and tools such as osnoise tracer and rtda osnoise [3,4] improved the CPU isolation analysis. Nevertheless, this is an ongoing process, and discussions are needed to speed up solutions for existing discussions and improve the existing tools and methods.

With this microconference we thus want to get together to discuss open problems, most notably: how to improve the identification of OS noise sources (expanding runtime tracing and/or code inspection tools), how to track them publicly (repo/DB) and how to tackle the sources of noise that have already been identified (e.g. [1,2,3]).

A non exhaustive list of potential topics is:

- OS noise profiling (format and public DB for the community)
- Tracing to detect OS noise: the rtda osnoise tracer and what it's missing
- TLB/icache flush deferral [1]
- Extend cpuset v2 cpu partition feature to replace isolcpus kernel cmdline
- rt-trace-bpf tool
- Task isolation
- smp\_call\_function API improvements

Potential attendees list:

Peter Zijlstra, Thomas Gleixner [tglx@linutronix.de](mailto:tglx@linutronix.de), Steven Rostedt, Juri Lelli, Daniel Bristot de Oliveira, Sebastian Andrzej Siewior, Valentin Schneider, Clark Williams, Paul E. McKenney, Paul Turner, Rik van Riel, Phil Auld, Waiman Long, Frederic Weisbecker, Nicolas Saenz Julienne, Marcelo Tosatti, Karl Rister, Andrew Theurer, Luis Claudio R. Gonçalves

MC leads:

Juri Lelli

Daniel Bristot de Oliveira

refs.

1 - <https://lore.kernel.org/all/20210929151723.162004989@infradead.org/>

2 - <https://lore.kernel.org/lkml/20220315153132.717153751@fedora.localdomain/>

3 - <https://lore.kernel.org/lkml/20211103170512.2745765-1-nsaenzju@redhat.com/>

4 - <https://www.kernel.org/doc/html/latest/trace/osnoise-tracer.html>

5 - <https://www.kernel.org/doc/html/latest/tools/rtla/rtla-osnoise.html>

**Primary authors:** LELLI, Juri (Red Hat); BRISTOT DE OLIVEIRA, Daniel (Red Hat, Inc.)

**Presenter:** BRISTOT DE OLIVEIRA, Daniel (Red Hat, Inc.)

**Track Classification:** LPC Microconference Track (CLOSED)