



Contribution ID: 75

Type: **not specified**

## Priority Inheritance for CFS Bandwidth Control

*Wednesday, 18 September 2024 16:07 (20 minutes)*

Throttling-like mechanisms such as CFS bandwidth control, extremely biased cgroup CPU shares and CPU masks can create quasi priorities among CFS tasks, and we can get priority inversion without explicit priority. We had such a problem caused by deep CPU throttling with CFS bandwidth control and it was causing application timeouts and down time.

To solve this problem we created a priority inheritance or priority ceiling like mechanism. The core idea of the solution is to treat the entire kernel mode as a critical section and not to throttle while in kernel mode. (There is an independently conceived, similar solution being discussed in lkml. We have posted the core part of our solution and they might merge - <https://lore.kernel.org/all/xm26edfxpock.fsf@bsegall-linux.svl.corp.google.com>)

Our solution not only reduced application timeouts, it also increased machine capacity. Each machine can now run a mix of workloads at higher CPU utilization without breaking down. We will discuss the solution, real-world data and data analysis.

**Primary author:** WANG, Xi

**Presenter:** WANG, Xi

**Session Classification:** Sched MC

**Track Classification:** Sched MC