Contribution ID: **62**

Type: **not specified**

”cat /proc/PID/maps”: What Could Possibly Go Wrong?

Thursday, 23 September 2021 07:45 (30 minutes)

Large installations require considerable monitoring and control, and the occasional scan of procfs files is often the best tool for the monitoring job at hand. In cases where memory consumption is a concern, /proc/PID/{maps,numa_maps,smaps,smaps_rollup} can be quite helpful.

To your monitoring, anyway.

Unfortunately, some mm-related procfs files need to acquire the dreaded mmap_sem. This can be a problem if the Very Important Process being monitored needs to modify its address space. Especially if your monitoring software has been fenced into a highly CPU-constrained cgroups-based container, in order to avoid interfering with Very Important Processes. Except that all of these procfs files acquire sleeplocks that might also be acquired by your Very Important Process. Plus your monitoring software might be preempted while holding one of these sleeplocks, that after all being the whole point of the aforementioned container. This can (and does) result in severe performance degradation.

Infrequently and intermittently.

We therefore have an abusive stress test that forces this condition to occur on small systems in less than one minute’s time [1].

This proposal, if accepted, will demonstrate this test program and a few schemes intended to make procfs-based monitoring safe for Very Important Processes [2].

[1] https://github.com/paulmckrcu/proc-mmap_sem-test
[2] https://git.infradead.org/users/willy/linux-maple.git/shortlog/refs/heads/proc-vma-rcu

I agree to abide by the anti-harassment policy

I agree

Primary authors: MCKENNEY, Paul (Facebook); WILCOX, Matthew (Oracle)

Presenters: MCKENNEY, Paul (Facebook); WILCOX, Matthew (Oracle)

Session Classification: Performance and Scalability MC

Track Classification: Performance and Scalability MC