Subject: [PATCH -mm 0/3] cgroup: block device i/o bandwidth controller (v7)
Posted by Andrea Righi on Tue, 22 Jul 2008 20:58:56 GMT
View Forum Message <> Reply to Message

The objective of the i/o bandwidth controller is to improve i/o performance predictability of different cgroups sharing the same block devices.

Respect to other priority/weight-based solutions the approach used by this controller is to explicitly choke applications' requests that directly (or indirectly) generate i/o activity in the system.

The direct bandwidth limiting method has the advantage of improving the performance predictability at the cost of reducing, in general, the overall performance of the system (in terms of throughput).

Detailed informations about design, its goal and usage are described in the documentation.

Tested against 2.6.26-rc8-mm1.

The all-in-one patch (and previous versions) can be found at:
http://download.systemimager.org/~arighi/linux/patches/io-throttle/

Changelog: (v6 -> v7)
- added i/o operations per second throttling
- fixed a build bug in x86 (undefined reference to `__udivdi3')
- updated documentation

Following some results of a simple test I did to check the effectiveness of the new iops throttling functionality (for Subrata: I'll post an update for the io-throttle testcase in LTP ASAP).

testcase overview
==========
- cgroup #1: process P1 periodically reads a 5.5MB file and prints in stdout the time needed to read the file

- cgroup #2: a process P2 is started; P2 runs a lot of parallel md5sums of all the files under /usr (recursively)

We want to improve P1 responsiveness and better predict P1 performance, regardless of the other i/o activities in the system, so we're going to measure the times printed by P1 in stdout to evaluate the effectiveness of a each tested solution for our particular requirement.

different configurations
==========
#1: no limiting at all

#2: plain CFQ priorities (P1 runs at real-time prio class 0, P2 runs at idle prio)
#3: iops throttling (P1 = unlimited, P2 = 50 iops/sec)
#4: bandwidth throttling (P1 = unlimited, P2 = 512KiB/s)
#5: bandwidth + iops throttling (P1 = unlimited, P2 = 512KiB/s and 50 iops/sec)
#6: aggressive bandwidth + iops throttling (P1 = unlimited, P2 = 128KiB/s and 10 iops/sec)

results (P1 response times)
==========

#1	#2	#3	#4	#5	#6
4.69724	4.68447	4.80822	4.37353	4.40609	4.37175
4.71427	4.45847	4.40524	4.35441	4.37228	4.35842
4.73120	4.46849	4.39400	4.36893	4.47388	4.36529
4.83120	4.47956	4.37878	4.44221	4.36823	4.37942
4.68060	4.49554	4.43058	4.40074	4.46004	4.37354
_____ P2 starts here! _____					
62.83110	7.06834	6.54557	7.10171	7.21964	5.35958
59.04400	6.92486	10.30330	5.38122	5.76458	4.89837
37.23380	7.11255	9.16971	8.32928	5.37017	5.51931
32.28180	7.26239	8.91513	6.27551	5.03347	4.79848
28.74150	7.19909	8.38274	5.00802	5.50771	4.72832

-Andrea

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers