Subject: Re: [PATCH 0/3] cgroup: block device i/o bandwidth controller (v3)
Posted by Subrata Modak on Fri, 20 Jun 2008 14:39:17 GMT

Dear Andrea,

We have been tracking Controllers developement for quite some time, and in order to make them work better, we have followed test-driven development for both CPU and Memory Controllers. We already have much of the CPU controllers and few Memory controllers testcases in LTP ( http://ltp.cvs.sourceforge.net/ltp/ltp/testcases/kernel/controllers/).

Test cases for Memory controllers, which are in 2.6.26-mm* tree(s), are also being readied. In the backdrop of this development, i would like to know if you can contribute to LTP test cases under GPLv2 for the i/o bandwidth controller, that you are proposing and any such future work in your pipeline. We would also be happy if you can look in to the existing controllers test cases, and give us more feedback on the same. Expecting to hear from you.

Regards--
Subrata
(LTP Maintainer)

On Fri, Jun 20, 2008 at 3:35 PM, Andrea Righi <righi.andrea@gmail.com> wrote:

>
> The goal of the i/o bandwidth controller is to improve i/o performance
> predictability and provide better QoS for 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 latest git (2.6.26-rc6).
>
> The all-in-one patch (and previous versions) can be found at:
>

http://download.systemimager.org/~arighi/linux/patches/io-throttle/<http://download.systemimager.org/%7Earighi/linux/patches/io-throttle/>
>
> Changelog: (v2 -> v3)
>  - scalability improvement: replaced the rbtree structure with a linked
> list
>    to store multiple per block device I/O limiting rules; this allows to
> use
>    RCU to protect the whole list structure, since the elements in the list
> are
>    supposed to change rarely (this also provides zero overhead for cgroups
>    that don't use any I/O limitation)
>  - improved user interface
>      - now it's possible to specify a suffix k, K, m, M, g, G to express
>        bandwidth values in KB/s, MB/s or GB/s
>      - current per block device I/O usage is reported in blockio.bandwidth
>  - renamed cgroup_io_account() in cgroup_io_throttle()
>  - updated the documentation
>
> TODO:
>  - implement I/O throttling using a token bucket algorithm, as suggested by
>    Carl Henrik Lunde, in addition to the current leaky bucket approach
>  - provide a modular interface to switch between different i/o throttling
>    algorithms at run-time
>
> -Andrea
> --
> To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at  http://vger.kernel.org/majordomo-info.html
> Please read the FAQ at  http://www.tux.org/lkml/
>


--
Regards & Thanks--
Subrata

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