Subject: Re: [RFC][v2][patch 0/12][CFQ-cgroup]Yet another I/O bandwidth controlling subsystem for CGroups bas
Posted by Ryo Tsuruta on Tue, 03 Jun 2008 08:15:35 GMT
View Forum Message <> Reply to Message

Hi Uchida-san,

> I report my tests.

I did a similar test to yours. I increased the number of I/Os
which are issued simultaneously up to 100 per cgroup.

Procedures:
- o Prepare 300 files which size is 250MB on 1 partition sdb3
- o Create three groups with priority 0, 4 and 7.
- o Run many processes issuing random direct I/O with 4KB data on each files in three groups.
  - #1 Run 25 processes issuing read I/O only per group.
  - #2 Run 100 processes issuing read I/O only per group.
- o Count up the number of I/Os which have done in 10 minutes.

The number of I/Os (percentage to total I/O)

| group priority | group 1 0(highest) | group 2 4 | group 3 7(lowest) | total I/Os |
|---|---|---|---|---|
| Estimate Performance | 61.5% | 30.8% | 7.7% | |
| #1 25procs | 52763(57%) | 30811(33%) | 9575(10%) | 93149 |
| #2 100procs | 24949(40%) | 21325(34%) | 16508(26%) | 62782 |

The result of test #1 is close to your estimation, but the result
of test #2 is not, the gap between the estimation and the result
increased.

In addition, I got the following message during test #2. Program
"ioload", our benchmark program, was blocked more than 120 seconds.
Do you see any problems?

INFO: task ioload:8456 blocked for more than 120 seconds.
"echo 0 > /proc/sys/kernel/hung_task_timeout_secs" disables this message.
ioload D 00000008 2772 8456 8419
f72eb740 00200082 c34862c0 00000008 c3565170 c35653c0 c2009d80
00000001
c1d1bea0 00200046 ffffffff f6ee039c 00000000 00000000 00000000
c2009d80

```
       018db000 00000000 f71a6a00 c0604fb6 00000000 f71a6bc8 c04876a4
       00000000
Call Trace:
 [<c0604fb6>] io_schedule+0x4a/0x81
 [<c04876a4>] __blockdev_direct_IO+0xa04/0xb54
 [<c04a3aa2>] ext2_direct_IO+0x35/0x3a
 [<c04a4757>] ext2_get_block+0x0/0x603
 [<c044ab81>] generic_file_direct_IO+0x103/0x118
 [<c044abe6>] generic_file_direct_write+0x50/0x13d
 [<c044b59e>] __generic_file_aio_write_nolock+0x375/0x4c3
 [<c046e571>] link_path_walk+0x86/0x8f
 [<c044a1e8>] find_lock_page+0x19/0x6d
 [<c044b73e>] generic_file_aio_write+0x52/0xa9
 [<c0466256>] do_sync_write+0xbf/0x100
 [<c042ca44>] autoremove_wake_function+0x0/0x2d
 [<c0413366>] update_curr+0x83/0x116
 [<c0605280>] mutex_lock+0xb/0x1a
 [<c04b653b>] security_file_permission+0xc/0xd
 [<c0466197>] do_sync_write+0x0/0x100
 [<c046695d>] vfs_write+0x83/0xf6
 [<c0466ea9>] sys_write+0x3c/0x63
 [<c04038de>] syscall_call+0x7/0xb
 [<c0600000>] print_cpu_info+0x27/0x92
 =======================
```

Thanks,
Ryo Tsuruta

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