
Subject: [PATCH 1/3] memcggroup: fix and update documentation

Posted by [Li Zefan](#) on Mon, 18 Feb 2008 05:59:24 GMT

[View Forum Message](#) <> [Reply to Message](#)

- remove trailing " Bytes"s in the demonstration
- remove section 4.4 (feature control_type has been removed)
- fix reference section

Signed-off-by: Li Zefan <lizf@cn.fujitsu.com>

Acked-by: KAMEZAWA Hiroyuki <kamezawa.hiroyu@jp.fujitsu.com>

Documentation/controllers/memory.txt | 24 ++++++++-----

1 files changed, 9 insertions(+), 15 deletions(-)

diff --git a/Documentation/controllers/memory.txt b/Documentation/controllers/memory.txt

index b5bbea9..6015347 100644

--- a/Documentation/controllers/memory.txt

+++ b/Documentation/controllers/memory.txt

@@ -170,14 +170,14 @@ NOTE: We can use a suffix (k, K, m, M, g or G) to indicate values in kilo,
mega or gigabytes.

cat /cgroups/0/memory.limit_in_bytes

-4194304 Bytes

+4194304

NOTE: The interface has now changed to display the usage in bytes instead of pages

We can check the usage:

cat /cgroups/0/memory.usage_in_bytes

-1216512 Bytes

+1216512

A successful write to this file does not guarantee a successful set of this limit to the value written into the file. This can be due to a

@@ -187,7 +187,7 @@ this file after a write to guarantee the value committed by the kernel.

echo -n 1 > memory.limit_in_bytes

cat memory.limit_in_bytes

-4096 Bytes

+4096

The memory.failcnt field gives the number of times that the cgroup limit was exceeded.

@@ -233,13 +233,6 @@ cgroup might have some charge associated with it, even though all tasks have migrated away from it. Such charges are automatically dropped at rmdir() if there are no tasks.

-4.4 Choosing what to account -- Page Cache (unmapped) vs RSS (mapped)?

-

-The type of memory accounted by the cgroup can be limited to just

-mapped pages by writing "1" to memory.control_type field

-

-echo -n 1 > memory.control_type

-

5. TODO

1. Add support for accounting huge pages (as a separate controller)

@@ -262,18 +255,19 @@ References

3. Emelianov, Pavel. Resource controllers based on process cgroups

<http://lkml.org/lkml/2007/3/6/198>

4. Emelianov, Pavel. RSS controller based on process cgroups (v2)

- <http://lkml.org/lkml/2007/4/9/74>

+ <http://lkml.org/lkml/2007/4/9/78>

5. Emelianov, Pavel. RSS controller based on process cgroups (v3)

<http://lkml.org/lkml/2007/5/30/244>

6. Menage, Paul. Control Groups v10, <http://lwn.net/Articles/236032/>

7. Vaidyanathan, Srinivasan, Control Groups: Pagecache accounting and control subsystem (v3), <http://lwn.net/Articles/235534/>

-8. Singh, Balbir. RSS controller V2 test results (Imbench),

+8. Singh, Balbir. RSS controller v2 test results (Imbench),

<http://lkml.org/lkml/2007/5/17/232>

-9. Singh, Balbir. RSS controller V2 AIM9 results

+9. Singh, Balbir. RSS controller v2 AIM9 results

<http://lkml.org/lkml/2007/5/18/1>

-10. Singh, Balbir. Memory controller v6 results,

+10. Singh, Balbir. Memory controller v6 test results,

<http://lkml.org/lkml/2007/8/19/36>

-11. Singh, Balbir. Memory controller v6, <http://lkml.org/lkml/2007/8/17/69>

+11. Singh, Balbir. Memory controller introduction (v6),

+ <http://lkml.org/lkml/2007/8/17/69>

12. Corbet, Jonathan, Controlling memory use in cgroups,

<http://lwn.net/Articles/243795/>

--

1.5.4.rc3

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>
