Subject: [-mm PATCH 5/8] Memory controller task migration (v2)
Posted by Balbir Singh on Fri, 06 Jul 2007 05:21:55 GMT
View Forum Message <> Reply to Message

Allow tasks to migrate from one container to the other. We migrate
mm_struct's mem_container only when the thread group id migrates.

Signed-off-by: Balbir Singh <balbir@linux.vnet.ibm.com>
---

 mm/memcontrol.c | 35 ++++++++++++++++++++++++++++++++++
 1 file changed, 35 insertions(+)

```
diff -puN mm/memcontrol.c~mem-control-task-migration mm/memcontrol.c
--- linux-2.6.22-rc6/mm/memcontrol.c~mem-control-task-migration 2007-07-05
13:45:18.000000000 -0700
+++ linux-2.6.22-rc6-balbir/mm/memcontrol.c 2007-07-05 13:45:18.000000000 -0700
@@ -302,11 +302,46 @@ err:
  return rc;
 }

+static void mem_container_move_task(struct container_subsys *ss,
+    struct container *cont,
+    struct container *old_cont,
+    struct task_struct *p)
+{
+ struct mm_struct *mm;
+ struct mem_container *mem, *old_mem;
+
+ mm = get_task_mm(p);
+ if (mm == NULL)
+  return;
+
+ mem = mem_container_from_cont(cont);
+ old_mem = mem_container_from_cont(old_cont);
+
+ if (mem == old_mem)
+  goto out;
+
+ /*
+  * Only thread group leaders are allowed to migrate, the mm_struct is
+  * in effect owned by the leader
+  */
+ if (p->tgid != p->pid)
+  goto out;
+
+ css_get(&mem->css);
```

```
+ rcu_assign_pointer(mm->mem_container, mem);
+ css_put(&old_mem->css);
+
+out:
+ mmput(mm);
+ return;
+}
+
 struct container_subsys mem_container_subsys = {
  .name = "mem_container",
  .subsys_id = mem_container_subsys_id,
  .create = mem_container_create,
  .destroy = mem_container_destroy,
  .populate = mem_container_populate,
+ .attach = mem_container_move_task,
  .early_init = 1,
 };
_

--
 Warm Regards,
 Balbir Singh
 Linux Technology Center
 IBM, ISTL
```

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