Subject: [PATCH 0/1] Add mount/umount callbacks to containers (Re: [ckrm-tech] [PATCH 4/6] containers: Simple
Posted by Balbir Singh on Mon, 15 Jan 2007 09:01:26 GMT
View Forum Message <> Reply to Message

Paul Menage wrote:
> On 1/10/07, Balbir Singh <balbir@in.ibm.com> wrote:
>> I have run into a problem running this patch on a powerpc box. Basically,
>> the machine panics as soon as I mount the container filesystem with
>
> This is a multi-processor system?
>
> My guess is that it's a race in the subsystem API that I've been
> meaning to deal with for some time - basically I've been using
> (<foo>_subsys.subsys_id != -1) to indicate that <foo> is ready for
> use, but there's a brief window during subsystem registration where
> that's not actually true.
>
> I'll add an "active" field in the container_subsys structure, which
> isn't set until registration is completed, and subsystems should use
> that instead. container_register_subsys() will set it just prior to
> releasing callback_mutex, and cpu_acct.c (and other subsystems) will
> check <foo>_subsys.active rather than (<foo>_subsys.subsys_id != -1)
>
>> I am trying to figure out the reason for the panic and trying to find
>> a fix. Since the introduction of whole hierarchy system, the debugging
>> has gotten a bit harder and taking longer, hence I was wondering if you
>> had any clues about the problem
>>
>

Hi, Paul,

I figured out the reason for the panic. Here are the fixes

Add mount and umount callbacks. These callbacks can be used by the
controller to figure out the correct root container and also know
whether the controller is currently acitve.

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

```
 include/linux/container.h |    2 ++
 kernel/container.c        |   12 ++++++++++++
 2 files changed, 14 insertions(+)

diff -puN include/linux/container.h~add-mount-callback include/linux/container.h
---  linux-2.6.20-rc3/include/linux/container.h~add-mount-callbac k 2007-01-12
```

```
21:23:00.000000000 +0530
+++ linux-2.6.20-rc3-balbir/include/linux/container.h 2007-01-12
21:23:00.000000000 +0530
@@ -171,6 +171,8 @@ struct container_subsys {
  void (*exit)(struct container_subsys *ss, struct task_struct *task);
  int (*populate)(struct container_subsys *ss,
    struct container *cont);
+ void (*mount)(struct container_subsys *ss, struct container *cont);
+ void (*umount)(struct container_subsys *ss, struct container *cont);

  int subsys_id;
 #define MAX_CONTAINER_TYPE_NAMELEN 32
diff -puN kernel/container.c~add-mount-callback kernel/container.c
--- linux-2.6.20-rc3/kernel/container.c~add-mount-callback 2007-01-12
21:23:00.000000000 +0530
+++ linux-2.6.20-rc3-balbir/kernel/container.c 2007-01-12 21:42:59.000000000 +0530
@@ -394,6 +394,7 @@ static void container_put_super(struct s
  int i;
  struct container *cont = &root->top_container;
  struct task_struct *g, *p;
+ struct container_subsys *ss;

  root->sb = NULL;
  sb->s_fs_info = NULL;
@@ -407,6 +408,11 @@ static void container_put_super(struct s

  mutex_lock(&callback_mutex);

+ for_each_subsys(hierarchy, ss) {
+  if (ss->umount)
+   ss->umount(ss, cont);
+ }
+
  /* Remove all tasks from this container hierarchy */
  read_lock(&tasklist_lock);
  do_each_thread(g, p) {
@@ -607,6 +613,12 @@ static int container_get_sb(struct file_
    rcu_assign_pointer(subsys[i]->hierarchy,
         hierarchy);
   }
+
+  for_each_subsys(hierarchy, ss) {
+   if (ss->mount)
+    ss->mount(ss, cont);
+  }
+
   mutex_unlock(&callback_mutex);
   synchronize_rcu();
```

--

Balbir Singh,
Linux Technology Center,
IBM Software Labs

PS: I hope my mailer does not word wrap the patches.