
Subject: Re: Re: [PATCH 2.6.24-rc8-mm1 09/15] (RFC) IPC: new kernel API to change an ID

Posted by [Pavel Emelianov](#) on Mon, 04 Feb 2008 14:06:23 GMT

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

Kirill Korotaev wrote:

>
> Cedric Le Goater wrote:
>> Hello Kirill !
>>
>> Kirill Korotaev wrote:
>>> Pierre,
>>>
>>> my point is that after you've added interface "set IPCID", you'll need
>>> more and more for checkpointing:
>>> - "create/setup conntrack" (otherwise connections get dropped),
>>> - "set task start time" (needed for Oracle checkpointing BTW),
>>> - "set some statistics counters (e.g. networking or taskstats)"
>>> - "restore inotify"
>>> and so on and so forth.
>> right. we know that we will have to handle a lot of these
>> and more and we will need an API for it :) so how should we handle it ?
>> through a dedicated syscall that would be able to checkpoint and/or
>> restart a process, an ipc object, an ipc namespace, a full container ?
>> will it take a fd or a big binary blob ?
>> I personally really liked Pavel idea's of filesystem. but we dropped the
>> thread.
>
> Imho having a file system interface means having all its problems.
> Imagine you have some information about tasks exported with a file system interface.
> Obviously to collect the information you have to hold some spinlock like tasklist_lock or similar.
> Obviously, you have to drop the lock between sys_read() syscalls.
> So interface gets much more complicated - you have to rescan the objects and somehow find
the place where
> you stopped previous read. Or you have to force reader to read everything at once.

To remember the place when we stopped previous read we have a "pos" counter
on the struct file.

Actually, tar utility, that I propose to perform the most simple migration
reads the directory contents with 4Kb buffer - that's enough for ~500 tasks.

Besides, is this a real problem for a frozen container?

>> that's for the user API but we will need also kernel services to expose
>> (checkpoint) states and restore them. If it's too
>> early to talk about the user API, we could try first to refactor
>> the kernel internals to expose correctly what we need.

>
> That's what I would start with.
>
>> That's what Pierre's patchset is trying to do.
>
> Not exactly. For checkpointing/restoring we actually need only one new API call for each
> subsystem - create some object with given ID (and maybe parameters, if they are not
> dynamically changeable by user).
> While Pierre's patchset adds different API call - change object ID.
>
> Thanks,
> Kirill
> _____
> Containers mailing list
> Containers@lists.linux-foundation.org
> <https://lists.linux-foundation.org/mailman/listinfo/containers>
>

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