
Subject: Re: [PATCH v7 09/10] IPC: message queue copy feature introduced
Posted by [Stanislav Kinsbursky](#) on Thu, 18 Oct 2012 11:34:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

> On Thu, Oct 18, 2012 at 1:02 PM, Stanislav Kinsbursky
> <skinsbursky@parallels.com> wrote:

>>
>>> On Thu, Oct 18, 2012 at 12:23 PM, Stanislav Kinsbursky
>>> <skinsbursky@parallels.com> wrote:
>>>>
>>>> This patch is required for checkpoint/restore in userspace.
>>>> IOW, c/r requires some way to get all pending IPC messages without
>>>> deleting
>>>> them from the queue (checkpoint can fail and in this case tasks will be
>>>> resumed,
>>>> so queue have to be valid).
>>>> To achive this, new operation flag MSG_COPY for sys_msgrcv() system call
>>>> was
>>>> introduced. If this flag was specified, then mtype is interpreted as
>>>> number of
>>>> the message to copy.
>>>> If MSG_COPY is set, then kernel will allocate dummy message with passed
>>>> size,
>>>> and then use new copy_msg() helper function to copy desired message
>>>> (instead of
>>>> unlinking it from the queue).
>>>>
>>>> Notes:
>>>> 1) Return -ENOSYS if MSG_COPY is specified, but CONFIG_CHECKPOINT_RESTORE
>>>> is
>>>> not set.
>>>
>>>
>>> Stanislav,
>>>
>>> A naive question, because I have not followed C/R closely. How do you
>>> deal with the case that other processes may be reading from the queue?
>>> (Or is that disabled during checkpointing?)
>>>
>>
>> To be honest, in this case behaviour in user-space is unpredictable.
>> I.e. if you have, for example, 5 messages in queue and going to peek them
>> all, and another process is reading the queue in the same time, then, most
>> probably, you won't peek all the 5 and receive ENOMSG.
>> But this case can be easily handled by user-space application (number of
>> messages in queue can be discovered before peeking).

>>
>> Note, that in CRIU IPC resources will be collected when all processes to
>> migrate are frozen.
>
> Perhaps I am missing something fundamental, but how can C/R sanely do
> anything at all here?
>
> For example, suppose a process reads and processes a message after you
> read it with MSG_COPY. Then the remaining messages are all shifted by
> one position, and you are going to miss reading one of them. IIUC the
> idea of MSG_COPY is to allow you to retrieve a copy of all messages in
> the list. It sounds like there's no way this can be done reliably. So,
> what possible use does the operation have?
>

First of all, this problem exist as is regardless to C/R feature or this patch set. If you share some resource (like message queue in this particular case) system-wide, then any process A can read out a message, which was send by process B to process C. So, when processes uses IPC message queues, they should be designed to handle such failures.

Second, it's up to user-space how to handle such things. It's implied, that user, trying to migrate some process, holding one end of queue, will also migrate another process, holding second end.

Third, there is IPC namespace, which isolates IPC objects. It can be used for safe migration of process tree.

> Thanks,
>
> Michael
>
>

--

Best regards,
Stanislav Kinsbursky
