Subject: [RFC][PATCH] Use task_pgrp()/task_session() in copy_process
Posted by Sukadev Bhattiprolu on Thu, 11 Jan 2007 15:58:16 GMT
View Forum Message <> Reply to Message

I am trying to replace process_group() and process_session() calls in
copy_process() with task_pgrp() and task_session().

Since task_pid() task_pgrp(), task_session() for the swapper are NULL, I
had to treat swapper as special in this patch and would like some comments.

Should we instead do some magic in start_kernel() so we don't have to treat
swapper special or maybe move the check for pid_t == 0 into task_pgrp()/
task_session() ?

P.S: For find_attach_pid()/attach_pid(), pls see my recent patches to Containers@
    and LKML.

From: Sukadev Bhattiprolu <sukadev@us.ibm.com>

Replace process_group() and process_session() with container-friendly task_pgrp() and
task_session().

Signed-off-by: Sukadev Bhattiprolu <sukadev@us.ibm.com>
Cc: Cedric Le Goater <clg@fr.ibm.com>
Cc: Dave Hansen <haveblue@us.ibm.com>
Cc: Serge Hallyn <serue@us.ibm.com>
Cc: containers@lists.osdl.org
---
 kernel/fork.c | 9 +++++++--
 1 files changed, 7 insertions(+), 2 deletions(-)

Index: lx26-20-rc2-mm1/kernel/fork.c
===
--- lx26-20-rc2-mm1.orig/kernel/fork.c 2007-01-11 07:18:03.383853328 -0800
+++ lx26-20-rc2-mm1/kernel/fork.c 2007-01-11 07:19:55.550801360 -0800
@@ -1248,8 +1248,13 @@ static struct task_struct *copy_process(
    p->signal->tty = current->signal->tty;
    p->signal->pgrp = process_group(current);
    set_signal_session(p->signal, process_session(current));
-   find_attach_pid(p, PIDTYPE_PGID, process_group(p));
-   find_attach_pid(p, PIDTYPE_SID, process_session(p));
+   if (current->pid) {
+   attach_pid(p, PIDTYPE_PGID, task_pgrp(current));
+   attach_pid(p, PIDTYPE_SID, task_session(current));
+   } else {
+   find_attach_pid(p, PIDTYPE_PGID, process_group(current));
+   find_attach_pid(p, PIDTYPE_SID, process_session(current));
+   }

list_add_tail_rcu(&p->tasks, &init_task.tasks);
__get_cpu_var(process_counts)++;

_____

Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers

Subject: Re: [RFC][PATCH] Use task_pgrp()/task_session() in copy_process
Posted by Dave Hansen on Thu, 11 Jan 2007 18:04:29 GMT
View Forum Message <> Reply to Message

On Thu, 2007-01-11 at 07:58 -0800, Sukadev Bhattiprolu wrote:
> ===
> --- lx26-20-rc2-mm1.orig/kernel/fork.c 2007-01-11 07:18:03.383853328 -0800
> +++ lx26-20-rc2-mm1/kernel/fork.c 2007-01-11 07:19:55.550801360 -0800
> @@ -1248,8 +1248,13 @@ static struct task_struct *copy_process(
> p->signal->tty = current->signal->tty;
> p->signal->pgrp = process_group(current);
> set_signal_session(p->signal, process_session(current));
> - find_attach_pid(p, PIDTYPE_PGID, process_group(p));
> - find_attach_pid(p, PIDTYPE_SID, process_session(p));
> + if (current->pid) {
> + attach_pid(p, PIDTYPE_PGID, task_pgrp(current));
> + attach_pid(p, PIDTYPE_SID, task_session(current));
> + } else {
> + find_attach_pid(p, PIDTYPE_PGID, process_group(current));
> + find_attach_pid(p, PIDTYPE_SID, process_session(current));
> + }
>
> list_add_tail_rcu(&p->tasks, &init_task.tasks);
> __get_cpu_var(process_counts)++;

I know I've asked this before (and I know I'm going to ask it again),
but why do we need both task_pgrp() and process_group() to both have
similar-sounding names and both take the same kind of argument? :) This
stuff _really_ needs to get cleaned up. It makes reviewing these
patches much harder.

In general, you should keep the hacks (which this is) to boot and
init-time stuff. If you can initialize a structure so that it plays
nicely for the rest of its life, do that. Don't put special cases in
common code that everybody will have to look at.

> Since task_pid() task_pgrp(), task_session() for the swapper are NULL, I
> had to treat swapper as special in this patch and would like some comments.

Can you do some research and find out _why_ these are NULL, and why they
need to be kept NULL?

-- Dave

_____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers

Subject: Re: [RFC][PATCH] Use task_pgrp()/task_session() in copy_process
Posted by Sukadev Bhattiprolu on Thu, 11 Jan 2007 19:44:23 GMT
View Forum Message <> Reply to Message

Dave Hansen [haveblue@us.ibm.com] wrote:
| On Thu, 2007-01-11 at 07:58 -0800, Sukadev Bhattiprolu wrote:
| > ===
| > --- lx26-20-rc2-mm1.orig/kernel/fork.c 2007-01-11 07:18:03.383853328 -0800
| > +++ lx26-20-rc2-mm1/kernel/fork.c 2007-01-11 07:19:55.550801360 -0800
| > @@ -1248,8 +1248,13 @@ static struct task_struct *copy_process(
| > p->signal->tty = current->signal->tty;
| > p->signal->pgrp = process_group(current);
| > set_signal_session(p->signal, process_session(current));
| > - find_attach_pid(p, PIDTYPE_PGID, process_group(p));
| > - find_attach_pid(p, PIDTYPE_SID, process_session(p));
| > + if (current->pid) {
| > + attach_pid(p, PIDTYPE_PGID, task_pgrp(current));
| > + attach_pid(p, PIDTYPE_SID, task_session(current));
| > + } else {
| > + find_attach_pid(p, PIDTYPE_PGID, process_group(current));
| > + find_attach_pid(p, PIDTYPE_SID, process_session(current));
| > + }
| >
| > list_add_tail_rcu(&p->tasks, &init_task.tasks);
| > __get_cpu_var(process_counts)++;
|
| I know I've asked this before (and I know I'm going to ask it again),
| but why do we need both task_pgrp() and process_group() to both have
| similar-sounding names and both take the same kind of argument? :) This
| stuff _really_ needs to get cleaned up. It makes reviewing these
| patches much harder.

We are phasing out process_group(), process_session() which return a
pid_t. I guess it also points to not having a special case for swapper.

|

| In general, you should keep the hacks (which this is) to boot and
| init-time stuff. If you can initialize a structure so that it plays
| nicely for the rest of its life, do that. Don't put special cases in
| common code that everybody will have to look at.
|
| > Since task_pid() task_pgrp(), task_session() for the swapper are NULL, I
| > had to treat swapper as special in this patch and would like some comments.
|
| Can you do some research and find out _why_ these are NULL, and why they
| need to be kept NULL?

task_struct for swapper is initialized by hand (INIT_TASK, INIT_SIGNALS
etc) but no struct pid is ever allocated and attached to the swapper.
This is normally done in copy_process() and so is done for all other
processes starting with pid_t = 1 (/sbin/init).

I am trying to understand if there is a history to it and if they need to
be kept NULL.

|
| -- Dave

_____

Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers

Subject: Re: [RFC][PATCH] Use task_pgrp()/task_session() in copy_process
Posted by ebiederm on Thu, 11 Jan 2007 20:54:19 GMT
View Forum Message <> Reply to Message

Sukadev Bhattiprolu <sukadev@us.ibm.com> writes:

> Dave Hansen [haveblue@us.ibm.com] wrote:
> |
> | I know I've asked this before (and I know I'm going to ask it again),
> | but why do we need both task_pgrp() and process_group() to both have
> | similar-sounding names and both take the same kind of argument? :) This
> | stuff _really_ needs to get cleaned up. It makes reviewing these
> | patches much harder.
> >
> We are phasing out process_group(), process_session() which return a
> pid_t. I guess it also points to not having a special case for
> swapper.

Definitely. Removing the special cases is good.

> | In general, you should keep the hacks (which this is) to boot and
> | init-time stuff. If you can initialize a structure so that it plays
> | nicely for the rest of its life, do that. Don't put special cases in
> | common code that everybody will have to look at.
> |
> | > Since task_pid() task_pgrp(), task_session() for the swapper are NULL, I
> | > had to treat swapper as special in this patch and would like some comments.
> |
> | Can you do some research and find out _why_ these are NULL, and why they
> | need to be kept NULL?
>
>
> task_struct for swapper is initialized by hand (INIT_TASK, INIT_SIGNALS
> etc) but no struct pid is ever allocated and attached to the swapper.
> This is normally done in copy_process() and so is done for all other
> processes starting with pid_t = 1 (/sbin/init).
>
> I am trying to understand if there is a history to it and if they need to
> be kept NULL.

When attach_pid has completed successfully as well as having a struct pid pointer in your task_struct you are also on the appropriate list of that struct pid. So you can be found for signal delivery. Preserving that property for the init_task would be nice but we don't have that property for any other kernel thread so it should not be a big deal to place it in session and process group 1 before the first fork. There are enough corner cases I don't think we can set it all up with static initializers though.

Largely I would suggest that we have enough information that if we are going to do this conversion we don't go through an intermediate step of find_attach_pid. There are few enough users we should just be able to do a handful of preparatory patches and just convert all of the uses of attach_pid.

As for the rest of the history struct pid happened since things started being placed in git so you can find out a lot of the history and context with a simple git-log.

Generally I take a fairly pragmatic approach. If I can't see a use for a change I don't send it. Which simply means attach_pid not taking a struct pid hasn't been a blocker for anything I have done lately. I think it makes sense to convert attach_pid.

I think leaving an attach_find_pid behind is a horrible idea. There are not enough callers of attach_pid to make that worthwhile.

set_special_pids can get it's pid from the init_task. Although we

need to kill daemonize in the kernel (or at the very least upgraded
it to support all of the namespaces we have merged).

sys_setsid already has a struct pid for it's session so it can call
__set_special_pids with that.

In de_thread we already have a struct pid.
In sys_setpgid we check to ensure the struct pid already exists.
And in fork we already have a struct pid everywhere except
that special init_task case.

So it probably makes sense for pidmap_init to initialize the
pid for the session and group of the idle task. And then there
are no special cases left.

Eric

_____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers

Subject: Re: [RFC][PATCH] Use task_pgrp()/task_session() in copy_process
Posted by ebiederm on Thu, 11 Jan 2007 21:19:28 GMT
View Forum Message <> Reply to Message

ebiederm@xmission.com (Eric W. Biederman) writes:
>
> So it probably makes sense for pidmap_init to initialize the
> pid for the session and group of the idle task. And then there
> are no special cases left.

Well that almost works except if we did that alloc_pid could not
successfully allocate pid 1. Grumble getting those special cases
out of the boot path is a pain.

If we had a non-hashed struct pid (init_pid?) that we filled in
early (statically?), that would keep copy_process happy.

Then we would need to call setsid() in the kernel right after
the fork to assign a legitimate session and process group
to pid == 1.

Since the idle thread is not doing anything it shouldn't matter,
although we can attach the idle thread after the fork to session
and process group == 1 or set them to NULL if there is a corner case
is anything that cares.

Eric

_____

Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers