Subject: [PATCH 3/5] Use task_pid() to find leader's pid
Posted by Sukadev Bhattiprolu on Sun, 15 Jul 2007 04:57:30 GMT
View Forum Message <> Reply to Message

Subject: [PATCH 3/5] Use task_pid() to find leader's pid

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

Use task_pid() to get leader's pid since find_pid() cannot be used
after detach_pid(). See comments in the code below for more details.

Signed-off-by: Sukadev Bhattiprolu <sukadev@us.ibm.com>
Acked-by: Pavel Emelianov <xemul@openvz.org>

Cc: Eric W. Biederman <ebiederm@xmission.com>
Cc: Cedric Le Goater <clg@fr.ibm.com>
Cc: Dave Hansen <haveblue@us.ibm.com>
Cc: Serge Hallyn <serue@us.ibm.com>
Cc: Herbert Poetzel <herbert@13thfloor.at>
---
 fs/exec.c | 9 ++++++++-
 1 file changed, 8 insertions(+), 1 deletion(-)

Index: lx26-22-rc6-mm1/fs/exec.c
===
--- lx26-22-rc6-mm1.orig/fs/exec.c 2007-07-13 13:12:01.000000000 -0700
+++ lx26-22-rc6-mm1/fs/exec.c 2007-07-13 13:12:13.000000000 -0700
@@ -905,10 +905,17 @@ static int de_thread(struct task_struct
    * The old leader becomes a thread of the this thread group.
    * Note: The old leader also uses this pid until release_task
    *   is called. Odd but simple and correct.
+   * Note: With multiple pid namespaces, active pid namespace of
+   *   a process is stored in its struct pid. The detach_pid
+   *   below frees the struct pid, so we will have no notion
+   *   of an active pid namespace until we complete the
+   *   subsequent attach_pid(). Which means - calls like
+   *   find_pid()/pid_to_nr() return NULL and cannot be used
+   *   between the detach_pid() and attach_pid() calls.
    */
   detach_pid(tsk, PIDTYPE_PID);
   tsk->pid = leader->pid;
- attach_pid(tsk, PIDTYPE_PID, find_pid(tsk->pid));
+ attach_pid(tsk, PIDTYPE_PID, task_pid(leader));
   transfer_pid(leader, tsk, PIDTYPE_PGID);
   transfer_pid(leader, tsk, PIDTYPE_SID);
   list_replace_rcu(&leader->tasks, &tsk->tasks);

_____
Containers mailing list

Subject: Re: [PATCH 3/5] Use task_pid() to find leader's pid
Posted by Oleg Nesterov on Sun, 15 Jul 2007 12:17:50 GMT
View Forum Message <> Reply to Message

Sukadev Bhattiprolu wrote:
>
> Use task_pid() to get leader's pid since find_pid() cannot be used
> after detach_pid(). See comments in the code below for more details.
>
> ...
>
> + * Note: With multiple pid namespaces, active pid namespace of
> + * a process is stored in its struct pid. The detach_pid
> + * below frees the struct pid, so we will have no notion
> + * of an active pid namespace until we complete the
> + * subsequent attach_pid(). Which means - calls like
> + * find_pid()/pid_to_nr() return NULL and cannot be used
> + * between the detach_pid() and attach_pid() calls.

I think both the changelog and the comment are confusing,

> detach_pid(tsk, PIDTYPE_PID);
> tsk->pid = leader->pid;
> - attach_pid(tsk, PIDTYPE_PID, find_pid(tsk->pid));
> + attach_pid(tsk, PIDTYPE_PID, task_pid(leader));

because the change itself looks like an obvious performance fix, even
we don't use multiple pid namespaces. I don't think it is good idea to
add a fat comment which doesn't match the current reality, and find_pid()
should be avoided anyway.

Stupid question: why do we need to put the pid namespace into the struct
pid? Isn't it better if the user of the struct pid should know its ns?
For example, if /proc does put_pid(), that pid should be from the active
namespace.

Sukadev, could you cc me if you do that kind of changes?

Oleg.

Subject: Re: [PATCH 3/5] Use task_pid() to find leader's pid
Posted by Sukadev Bhattiprolu on Mon, 16 Jul 2007 19:59:52 GMT
View Forum Message <> Reply to Message

Oleg Nesterov [oleg@tv-sign.ru] wrote:
| Sukadev Bhattiprolu wrote:
| >
| > Use task_pid() to get leader's pid since find_pid() cannot be used
| > after detach_pid(). See comments in the code below for more details.
| >
| > ...
| >
| > +   * Note: With multiple pid namespaces, active pid namespace of
| > +   *   a process is stored in its struct pid. The detach_pid
| > +   *   below frees the struct pid, so we will have no notion
| > +   *   of an active pid namespace until we complete the
| > +   *   subsequent attach_pid(). Which means - calls like
| > +   *   find_pid()/pid_to_nr() return NULL and cannot be used
| > +   *   between the detach_pid() and attach_pid() calls.
|
| I think both the changelog and the comment are confusing,
|
| >   detach_pid(tsk, PIDTYPE_PID);
| >   tsk->pid = leader->pid;
| > - attach_pid(tsk, PIDTYPE_PID, find_pid(tsk->pid));
| > + attach_pid(tsk, PIDTYPE_PID, task_pid(leader));
|
| because the change itself looks like an obvious performance fix, even
| we don't use multiple pid namespaces. I don't think it is good idea to
| add a fat comment which doesn't match the current reality, and find_pid()
| should be avoided anyway.

Its a performance fix but also a correctness issue with multiple pid
namespaces. Here is the modified patch with the simplified changelog and
comment removed.

|
| Stupid question: why do we need to put the pid namespace into the struct
| pid? Isn't it better if the user of the struct pid should know its ns?
| For example, if /proc does put_pid(), that pid should be from the active
| namespace.

Not sure I fully understand this. A process, and by extension its 'struct
pid' is visible in multiple namespaces and we maintain this list of
namespaces in each 'struct pid'.

Are you suggesting having a pid_namespace with a list of all 'struct pids'
that are visible in it ?

|
| Sukadev, could you cc me if you do that kind of changes?

Sure - I will.

Subject: [PATCH 3/5] Use task_pid() to find leader's pid

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

Use task_pid() to get leader's 'struct pid' and avoid the find_pid().

Signed-off-by: Sukadev Bhattiprolu <sukadev@us.ibm.com>
Acked-by: Pavel Emelianov <xemul@openvz.org>

Cc: Eric W. Biederman <ebiederm@xmission.com>
Cc: Cedric Le Goater <clg@fr.ibm.com>
Cc: Dave Hansen <haveblue@us.ibm.com>
Cc: Serge Hallyn <serue@us.ibm.com>
Cc: Herbert Poetzel <herbert@13thfloor.at>
---
 fs/exec.c | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

Index: lx26-22-rc6-mm1a/fs/exec.c
==========
--- lx26-22-rc6-mm1a.orig/fs/exec.c 2007-07-13 18:23:55.000000000 -0700
+++ lx26-22-rc6-mm1a/fs/exec.c 2007-07-16 12:56:22.000000000 -0700
@@ -908,7 +908,7 @@ static int de_thread(struct task_struct
    */
   detach_pid(tsk, PIDTYPE_PID);
   tsk->pid = leader->pid;
- attach_pid(tsk, PIDTYPE_PID, find_pid(tsk->pid));
+ attach_pid(tsk, PIDTYPE_PID, task_pid(leader));
   transfer_pid(leader, tsk, PIDTYPE_PGID);
   transfer_pid(leader, tsk, PIDTYPE_SID);
   list_replace_rcu(&leader->tasks, &tsk->tasks);

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

---

Subject: Re: [PATCH 3/5] Use task_pid() to find leader's pid
Posted by Oleg Nesterov on Tue, 17 Jul 2007 13:44:15 GMT
View Forum Message <> Reply to Message

On 07/16, sukadev@us.ibm.com wrote:
>
> Oleg Nesterov [oleg@tv-sign.ru] wrote:
> |
> | Stupid question: why do we need to put the pid namespace into the struct
> | pid? Isn't it better if the user of the struct pid should know its ns?
> | For example, if /proc does put_pid(), that pid should be from the active
> | namespace.
>
> Not sure I fully understand this. A process, and by extension its 'struct
> pid' is visible in multiple namespaces and we maintain this list of
> namespaces in each 'struct pid'.
>
> Are you suggesting having a pid_namespace with a list of all 'struct pids'
> that are visible in it ?

I thought that the plan is: if the task is visible in some namespace, it has
a separate pid_t in that namespace.

OK, the question was relly stupid, please ignore. I'll wait for other patches
to understand what's going on.

Oleg.