Subject: Re: [PATCH 7/16] Helpers to find the task by its numerical ids
Posted by Pavel Emelianov on Tue, 10 Jul 2007 06:47:09 GMT
View Forum Message <> Reply to Message

sukadev@us.ibm.com wrote:
> Pavel Emelianov [xemul@openvz.org] wrote:
> | When searching the task by numerical id on may need to find
> | it using global pid (as it is done now in kernel) or by its
> | virtual id, e.g. when sending a signal to a task from one
> | namespace the sender will specify the task's virtual id.
> |
> | Signed-off-by: Pavel Emelianov <xemul@openvz.org>
> |
> | ---
> |
> | fs/proc/base.c | 2 +-
> | include/linux/pid.h | 13 ++++++++++--
> | include/linux/sched.h | 31 ++++++++++++++++++++++++++++++--
> | kernel/pid.c | 32 ++++++++++++++++++++++----------
> | 4 files changed, 58 insertions(+), 20 deletions(-)
> |
> | --- ./fs/proc/base.c.ve6 2007-07-06 10:58:56.000000000 +0400
> | +++ ./fs/proc/base.c 2007-07-06 11:03:41.000000000 +0400
> | @@ -2230,7 +2230,7 @@ static struct task_struct *next_tgid(uns
> | rcu_read_lock();
> | retry:
> | task = NULL;
> | - pid = find_ge_pid(tgid);
> | + pid = find_ge_pid(tgid, &init_pid_ns);
> | if (pid) {
> | tgid = pid->nr + 1;
> | task = pid_task(pid, PIDTYPE_PID);
> | --- ./include/linux/pid.h.ve6 2007-07-06 11:03:27.000000000 +0400
> | +++ ./include/linux/pid.h 2007-07-06 11:03:27.000000000 +0400
> | @@ -98,14 +98,23 @@ extern struct pid_namespace init_pid_ns;
> | /*
> | * look up a PID in the hash table. Must be called with the tasklist_lock
> | * or rcu_read_lock() held.
> | + *
> | + * find_pid_ns() finds the pid in the namespace specified
> | + * find_pid() find the pid by its global id, i.e. in the init namespace
> | + * find_vpid() finr the pid by its virtual id, i.e. in the current namespace
> | + *
> | + * see also find_task_by_pid() set in include/linux/sched.h
> | */
> | -extern struct pid *FASTCALL(find_pid(int nr));
> | +extern struct pid *FASTCALL(find_pid_ns(int nr, struct pid_namespace *ns));
> | +

> | +#define find_vpid(pid) find_pid_ns(pid, current->nsproxy->pid_ns)
> | +#define find_pid(pid) find_pid_ns(pid, &init_pid_ns)
>
> Adding a second interface maybe more confusing to drivers and non-pid
> users.
>
> But more importantly, modifying find_pid() to refer to only init_pid_ns
> would require auditing existing find_pid() callers and switching them to
> find_vpid().
>
> For instance if capset() is called from a child pid namespace, the 'pid'
> would refer to the pid or pgid from child pid ns. But cap_set_pg() calls
> find_pid() which gets the number from init_pid_ns.
>
> Is there a similar issue with sunos_killpg() ?
>

Yes, I know this. The [PATCH 15/16] has to switch all the kernel-to-user
boundaries to use the additional helpers. That's the hardest part and
I agree that I could lost something in it.

However, this is relevant only (!) when you clone the namespace. So people
who do not need them won't suffer when this patch set is in mainline.

That's my intention - to make a set that doesn't affect the non-namespace-d
case and go on polishing it. You have already pointed out 2 places. I expect
people to find more of them. This is easier to patch only the boundary to
the user rather than the whole kernel :)

Thanks,
Pavel