Subject: Re: [RFC][PATCH 06/16] Define is_global_init()
Posted by Sukadev Bhattiprolu on Thu, 24 May 2007 09:10:52 GMT
View Forum Message <> Reply to Message

Pavel Emelianov [xemul@sw.ru] wrote:
| > Index: lx26-21-mm2/kernel/pid.c
| > ==========
| > --- lx26-21-mm2.orig/kernel/pid.c 2007-05-22 16:59:34.000000000 -0700
| > +++ lx26-21-mm2/kernel/pid.c 2007-05-22 16:59:46.000000000 -0700
| > @@ -71,6 +71,27 @@ struct pid_namespace init_pid_ns = {
| > .child_reaper = &init_task
| > };
| >
| > +
| > +/**
| > + * is_global_init - check if a task structure is init
| > + * @tsk: Task structure to be checked.
| > + *
| > + * Check if a task structure is the first user space task the kernel created.
| > + */
| > +int is_global_init(struct task_struct *tsk)
| > +{
| > + return (task_active_pid_ns(tsk) == &init_pid_ns && tsk->pid == 1);
|
| This can OOPS if you pass arbitrary task to this call...
| tsk->nsproxy can already be NULL.

Hmm. You are right. btw, this could be a bisect issue. Patch 9 of uses
pid_ns from pid->upid_list and removes nsproxy->pid_ns.

|
| > +}
| > +
| > +/*
| > + * is_container_init:
| > + * check whether in the task is init in it's own pid namespace.
| > + */
| > +int is_container_init(struct task_struct *tsk)
| > +{
| > + return tsk->pid == 1;
| > +}
| > +
| > /*
| > * Note: disable interrupts while the pidmap_lock is held as an
| > * interrupt might come in and do read_lock(&tasklist_lock).
| > _____
| > Containers mailing list
| > Containers@lists.linux-foundation.org

| > https://lists.linux-foundation.org/mailman/listinfo/containers
| >
| > _____
| > Devel mailing list
| > Devel@openvz.org
| > https://openvz.org/mailman/listinfo/devel
| >

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