Subject: Re: [PATCH] allow a task to join a pid namespace
Posted by Oleg Nesterov on Mon, 04 Jun 2012 16:51:17 GMT
View Forum Message <> Reply to Message

On 06/04, Glauber Costa wrote:
>
> Currently, it is possible for a process to join existing
> net, uts and ipc namespaces. This patch allows a process to join an
> existing pid namespace as well.

I can't understand this patch... but probably I missed something,
I never really understood setns.

```
> +static int pidns_install(struct nsproxy *nsproxy, void *_ns)
> +{
> + struct pid *newpid;
> + struct pid_namespace *ns = _ns;
> +
> + if (is_container_init(current))
> +  return -EINVAL;
> +
> + if (nsproxy->pid_ns != ns->parent)
> +  return -EPERM;
```

At least you should also check that current is single-threaded,
I guess.

```
> +
> + if (task_pgrp(current) !=  task_pid(current))
> +  return -EPERM;
> +
> + if (task_session(current) !=  task_pid(current))
> +  return -EPERM;
```

Both checks are obviously racy without tasklist.

```
> + newpid = alloc_pid(ns);
> + if (!newpid)
> +  return -ENOMEM;
```

Hmm. Doesn't this mean that pid_nr of this task (as it seen
in its current namespace) will be changed? This doesn't look
sane.

```
> + put_pid_ns(nsproxy->pid_ns);
> + nsproxy->pid_ns = get_pid_ns(ns);
> +
> + write_lock_irq(&tasklist_lock);
> + change_pid(current, PIDTYPE_PID, newpid);
> + change_pid(current, PIDTYPE_PGID, newpid);
> + change_pid(current, PIDTYPE_SID, newpid);
> + write_unlock_irq(&tasklist_lock);
```

Hmm. So, until the caller does switch_task_namespaces()
task_active_pid_ns(current) != current->nsproxy->pid_ns,
doesn't look very nice too.

I don't think this can be right. If nothing else, this breaks
it_real_fn().

Oleg.