Subject: Re: [PATCH 0/16] Pid namespaces
Posted by Pavel Emelianov on Tue, 10 Jul 2007 07:02:51 GMT
View Forum Message <> Reply to Message

sukadev@us.ibm.com wrote:
> I am not able to find a specific patch that this might be in,
> but what happens when the child-reaper of a container exits ?

The init namespace's init becomes this init's namespace's init :)
In other words:

```
	if (unlikely(tsk == child_reaper(tsk))) {
		if (tsk->nsproxy->pid_ns != &init_pid_ns)
			tsk->nsproxy->pid_ns->child_reaper = init_pid_ns.child_reaper;
		else
			panic("Attempted to kill init!");
	}
```

> Do you terminate all processes in the container ? I thought
> that was discussed earlier and the consensus was to terminate
> all processes in that container and its subordinate containers.
>
> Is that not the case now ?

That's the case, but this code works without this change. We can do
it later.

> Suka
>
> Pavel Emelianov [xemul@openvz.org] wrote:
> | This is "submition for inclusion" of hierarchical, not kconfig
> | configurable, zero overheaded ;) pid namespaces.
> |
> | The overall idea is the following:
> |
> | The namespace are organized as a tree - once a task is cloned
> | with CLONE_NEWPIDS (yes, I've also switched to it :) the new
> | namespace becomes the parent's child and tasks living in the
> | parent namespace see the tasks from the new one. The numerical
> | ids are used on the kernel-user boundary, i.e. when we export
> | pid to user we show the id, that should be used to address the
> | task in question from the namespace we're exporting this id to.
> |
> | The main difference from Suka's patches are the following:
> |
> | 0. Suka's patches change the kernel/pid.c code too heavy.
> | This set keeps the kernel code look like it was without
> | the patches. However, this is a minor issue. The major is:

```
> |
> | 1. Suka's approach is to remove the notion of the task's
> |    numerical pid from the kernel at all. The numbers are
> |    used on the kernel-user boundary or within the kernel but
> |    with the namespace this nr belongs to. This results in
> |    massive changes of struct's members fro int pid to struct
> |    pid *pid, task->pid becomes the virtual id and so on and
> |    so forth.
> |    My approach is to keep the good old logic in the kernel.
> |    The task->pid is a global and unique pid, find_pid() finds
> |    the pid by its global id and so on. The virtual ids appear
> |    on the user-kernel boundary only. Thus drivers and other
> |    kernel code may still be unaware of pids unless they do not
> |    communicate with the userspace and get/put numerical pids.
> |
> | And some more minor differences:
> |
> | 2. Suka's patches have the limit of pid namespace nesting.
> |    My patches do not.
> |
> | 3. Suka assumes that pid namespace can live without proc mount
> |    and tries to make the code work with pid_ns->proc_mnt change
> |    from NULL to not-NULL from times to times.
> |    My code calls the kern_mount() at the namespace creation and
> |    thus the pid_namespace always works with proc.
> |
> | There are some small issues that I can describe if someone is
> | interested.
> |
> | The tests like nptl perf, unixbench spawn, getpid and others
> | didn't reveal any performance degradation in init_namespace
> | with the RHEL5 kernel .config file. I admit, that different
> | .config-s may show that patches hurt the performance, but the
> | intention was *not* to make the kernel work worse with popular
> | distributions.
> |
> | This set has some ways to move forward, but this is some kind
> | of a core, that do not change the init_pid_namespace behavior
> | (checked with LTP tests) and may require some hacking to do
> | with the namespaces only.
> |
> | Patches apply to 2.6.22-rc6-mm1.
> -
> To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
> the body of a message to majordomo@vger.kernel.org
> More majordomo info at  http://vger.kernel.org/majordomo-info.html
> Please read the FAQ at  http://www.tux.org/lkml/
>
```

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