
Subject: Re: [PATCH 16/28] [FLAT 1/6] Changes in data structures for flat model
Posted by [Pavel Emelianov](#) on Wed, 20 Jun 2007 10:53:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

Eric W. Biederman wrote:

> Pavel Emelianov <xemul@openvz.org> writes:

>

[snip]

```
>>> | --- ./include/linux/sched.h.flatdatast 2007-06-15 15:14:33.000000000 +0400
>>> | +++ ./include/linux/sched.h 2007-06-15 15:19:14.000000000 +0400
>>> | @@ -482,7 +482,10 @@ struct signal_struct {
>>> |   pid_t session __deprecated;
>>> |   pid_t __session;
>>> |   };
>>> | -
>>> | +#ifdef CONFIG_PID_NS_FLAT
>>> | + pid_t vpgrp;
>>> | + pid_t vsession;
>>> | +#endif
>>> | /* boolean value for session group leader */
>>> | int leader;
>>> |
>>> | @@ -944,6 +947,11 @@ struct task_struct {
>>> |   unsigned did_exec:1;
>>> |   pid_t pid;
>>> |   pid_t tgid;
>>> | +#ifdef CONFIG_PID_NS_FLAT
>>> | + /* hash the virtual ids as well */
>>> | + pid_t vpid;
>>> | + pid_t vtgid;
>>> | +#endif
```

>

> Adding vpgrp, vsession, vpid, and vtgid is wrong.

>

> A case can probably be made for caching the common case (users view),
> but we already have fields for that.

>

> For a global view we must use struct pid *, otherwise we are just asking
> for trouble.

Nope. If we have global unique numerical pid we're not asking for trouble. We're just making kernel work like it always did. Virtual pid makes sense *only* when interacting with user level.

Making task->pid virtual is asking for trouble.

> Eric

Pavel

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