
Subject: Re: [ckrm-tech] [PATCH 0/2] resource control file system - aka containers on top of nsproxy!

Posted by [Paul Menage](#) on Thu, 08 Mar 2007 09:10:27 GMT

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

On 3/7/07, Sam Vilain <sam@vilain.net> wrote:

>
> Ok, they share this characteristic with namespaces: that they group
> processes. So, they conceptually hang off task_struct. But we put them
> on ns_proxy because we've got this vague notion that things might be
> better that way.

Remember that I'm not the one pushing to move them into ns_proxy.
These patches are all Srivatsa's work. Despite that fact that they say
"Signed-off-by: Paul Menage", I'd never seen them before they were
posted to LKML, and I'm not sure that they're the right approach.
(Although some form of unification might be good).

>
> >> about this you still insist on calling this sub-system specific stuff
> >> the "container",
> >>
> > Uh, no. I'm trying to call a *grouping* of processes a container.
> >
>
> Ok, so is this going to supplant the namespaces too?

I don't know. It would be nice to have a single object hanging off the
task struct that contains all the various grouping pointers. Having
something that was flexible enough to handle all the required
behaviours, or else allowing completely different behaviours for
different subsets of that structure, could be the fiddly bit.

See my expanded reply to Eric' earlier post for a possible way of
unifying them, and simplifying the nsproxy and container.c code in the
process.

>

Resource groups isn't a terrible name for them (although I'd be
wondering whether the BeanCounters folks would object :-)) but the
intention is that they're more generic than purely for resource
accounting. (E.g. see my other email where I suggested that things
like task->mempolicy and task->user could potentially be treated in
the same way)

Task Group is a good name, except for the fact that it's too easily

confused with process group.

>
> And do we bother changing IPC namespaces or let that one slide?
>

I think that "namespace" is a fine term for the IPC id virtualization/restriction that ipc_ns provides. (Unless I'm totally misunderstanding the concept).

Paul

Containers mailing list
Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>

Subject: Re: [ckrm-tech] [PATCH 0/2] resource control file system - aka containers on top of nsproxy!
Posted by [serue](#) on Fri, 09 Mar 2007 16:50:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

Quoting Paul Menage (menage@google.com):
> On 3/7/07, Sam Vilain <sam@vilain.net> wrote:
> >
> >Ok, they share this characteristic with namespaces: that they group
> >processes.

Namespaces have a side effect of grouping processes, but a namespace is not defined by 'grouping proceses.' A container is, in fact, a group of processes.

> > So, they conceptually hang off task_struct. But we put them
> >on ns_proxy because we've got this vague notion that things might be
> >better that way.
>
> Remember that I'm not the one pushing to move them into ns_proxy.
> These patches are all Srivatsa's work. Despite that fact that they say
> "Signed-off-by: Paul Menage", I'd never seen them before they were
> posted to LKML, and I'm not sure that they're the right approach.
> (Although some form of unification might be good).

The nsproxy container subsystem could be said to be that unification. If we really wanted to I suppose we could now always mount the nsproxy subsystem, get rid of task->nsproxy, and always get thta through it's nsproxy subsystem container. But then that causes trouble with being able to mount a hierarachy like

```
mount -t container -o ns,cpuset
```

so we'd have to fix something. It also slows things down...

```
> >>> about this you still insist on calling this sub-system specific stuff
> >>> the "container",
> >>>
> >> Uh, no. I'm trying to call a *grouping* of processes a container.
> >>
> >
> >Ok, so is this going to supplant the namespaces too?
>
> I don't know. It would be nice to have a single object hanging off the
> task struct that contains all the various grouping pointers. Having
```

The namespaces aren't grouping pointers, they are resource id tables.

I stand by my earlier observation that placing namespace pointers and grouping pointers in the same structure means that pointer will end up pointing to itself.

```
> something that was flexible enough to handle all the required
> behaviours, or else allowing completely different behaviours for
> different subsets of that structure, could be the fiddly bit.
>
> See my expanded reply to Eric' earlier post for a possible way of
> unifying them, and simplifying the nsproxy and container.c code in the
> process.
```

Doesn't ring a bell, I'll have to look around for that...

```
> >
> > - resource groups (I get a strange feeling of d?j? v? there)
>
> Resource groups isn't a terrible name for them (although I'd be
```

I still like 'rug' for resource usage groups :)

-serge

Containers mailing list
Containers@lists.osdl.org
<https://lists.osdl.org/mailman/listinfo/containers>

Subject: Re: [ckrm-tech] [PATCH 0/2] resource control file system - aka containers

on top of nsproxy!

Posted by [Srivatsa Vaddagiri](#) on Thu, 22 Mar 2007 14:08:14 GMT

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

On Fri, Mar 09, 2007 at 10:50:17AM -0600, Serge E. Hallyn wrote:

> The nsproxy container subsystem could be said to be that unification.
> If we really wanted to I suppose we could now always mount the nsproxy
> subsystem, get rid of tsk->nsproxy, and always get thta through it's
> nsproxy subsystem container. But then that causes trouble with being
> able to mount a hierarachy like
>
> mount -t container -o ns,cpuset

What troubles will mounting both cpuset and ns in the same hierarchy cause? IMO that may be a good feature by itself, which makes it convenient to bind different containers to different cpusets.

In this case, we want 'ns' subsystem to override all decisions wrt mkdir of directories and also movement of tasks b/n different groups. This is automatically accomplished in the patches, by having ns subsystem veto mkdir/can_attach request which aren't allowed as per namespace semantics (but which may be allowed as per cpuset semantics).

> so we'd have to fix something. It also slows things down...

--

Regards,
vatsa

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