Subject: Pid namespace patchsets review
Posted by Sukadev Bhattiprolu on Sat, 10 Mar 2007 03:55:12 GMT
View Forum Message <> Reply to Message

Hi

I am sending out three sets of patches for review/comments on the
overall approach/design of the pid namespace.

First set is actually a single, trivial patch that defines/uses wrappers
to get/set the pid namespace of a task.

The second PATCHSET of 6 patches attaches a list of struct pid_nrs which
allow unsharing of pid namespace and thus allow a process to have a pid_t
value, one in each pid namespace.

The third PATCHSET of 5 patches attempt to decouple pid namespace from
nsproxy and allow us to exit pid_namespace independent of other namespaces
(i.e a more complete fix for the nfsd exit problem we ran into a few weeks
ago).

Appreciate your feedback/comments.

Note: The patches depend on a few other pid namespace patches that we already
looked at on Containers list in the recent past (mostly the patchset
to statically initialize a struct pid for swapper).

If you want to apply the patches and try out the clone/unshare, please
let me know and I can point you to a tarball of the complete quilt
patchset to apply on 2.6.20-mm2 and our simple unit test program.

Suka

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

Subject: Re: Pid namespace patchsets review
Posted by ebiederm on Sat, 10 Mar 2007 06:05:43 GMT
View Forum Message <> Reply to Message

It is good to see these patches are starting to come together.

Be patient a good review is going to take me a little bit.

A couple of immediate things I see that would be nice to address before
we aim at merging these patches upstream.

- Since there are known cases that we still need to convert to use struct
  pid can we disable the clone/unshare unless we have the CONFIG_EXPERIMENTAL
  flag set. And a comment in Kconfig saying we are almost but not quite
  there yet. With that in place I would have no problems with the idea
  of merging all of the bits needed to have multiple pid namespaces before
  we finish making the code pid namespace safe.

- When we do the rename can we please rename it task_proxy and have the functions
  follow that naming. The resource limiting conversation seems to be going in
  that direction, and it more general then what we are using now.

- At a first skim the patches didn't quite feel like they were git-bisect safe.
  I haven't looked closely enough to be certain yet.

Eric

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

Subject: Re: Pid namespace patchsets review
Posted by Sukadev Bhattiprolu on Sat, 10 Mar 2007 18:24:05 GMT
View Forum Message <> Reply to Message

Eric W. Biederman [ebiederm@xmission.com] wrote:
|
| It is good to see these patches are starting to come together.
|
| Be patient a good review is going to take me a little bit.

Ok.
|
| A couple of immediate things I see that would be nice to address before
| we aim at merging these patches upstream.
|
| - Since there are known cases that we still need to convert to use struct
| pid can we disable the clone/unshare unless we have the CONFIG_EXPERIMENTAL
| flag set. And a comment in Kconfig saying we are almost but not quite
| there yet. With that in place I would have no problems with the idea
| of merging all of the bits needed to have multiple pid namespaces before
| we finish making the code pid namespace safe.

Agree.

|

| - When we do the rename can we please rename it task_proxy and have the functions
| follow that naming. The resource limiting conversation seems to be going in
| that direction, and it more general then what we are using now.

Agree.

|
| - At a first skim the patches didn't quite feel like they were git-bisect safe.
| I haven't looked closely enough to be certain yet.

Yes. They were safe until my most recent changes :-) We are working on
cleaning that up.
|
|
| Eric

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

Subject: Re: Pid namespace patchsets review
Posted by Herbert Poetzl on Sat, 10 Mar 2007 22:05:59 GMT
View Forum Message <> Reply to Message

On Fri, Mar 09, 2007 at 11:05:43PM -0700, Eric W. Biederman wrote:
>
> It is good to see these patches are starting to come together.
>
> Be patient a good review is going to take me a little bit.
>
> A couple of immediate things I see that would be nice to address before
> we aim at merging these patches upstream.
>
> - Since there are known cases that we still need to convert to use
>   struct pid can we disable the clone/unshare unless we have the
>   CONFIG_EXPERIMENTAL flag set. And a comment in Kconfig saying we
>   are almost but not quite there yet. With that in place I would have
>   no problems with the idea of merging all of the bits needed to have
>   multiple pid namespaces before we finish making the code pid namespace
>   safe.

IMHO not the best idea, mainly because both OpenVZ
and Linux-VServer will end up either duplicating
the pid code or using the incomplete (broken) version
which probably gives the pid space a bad start ...

I'd prefer to focus on fixing up the existing pid

issues (conversion) first, then hitting it with a
hopefully working pid namespace ...

YMMV

> - When we do the rename can we please rename it task_proxy and have
> the functions follow that naming. The resource limiting conversation
> seems to be going in that direction, and it more general then what we
> are using now.

hmm, nsproxy was unusual but kind of understandable,
task_proxy sounds just weird to me, I'd definitely
prefer nsproxy over task_proxy, but I'm open for
more 'space' related names too, like spaces or
space_proxy or space_group ...

best,
Herbert

> - At a first skim the patches didn't quite feel like they were
> git-bisect safe.
> I haven't looked closely enough to be certain yet.
>
>
> Eric
> _____
> Containers mailing list
> Containers@lists.osdl.org
> https://lists.osdl.org/mailman/listinfo/containers

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

Subject: Re: Pid namespace patchsets review
Posted by ebiederm on Sun, 11 Mar 2007 01:57:13 GMT
View Forum Message <> Reply to Message

Herbert Poetzl <herbert@13thfloor.at> writes:

> IMHO not the best idea, mainly because both OpenVZ
> and Linux-VServer will end up either duplicating
> the pid code or using the incomplete (broken) version
> which probably gives the pid space a bad start ...
>
> I'd prefer to focus on fixing up the existing pid
> issues (conversion) first, then hitting it with a

> hopefully working pid namespace ...
>
> YMMV

Right now if we discount the kernel_thread to kthread conversion
we are probably 98% done with all of the conversions that make sense
without a pid namespace.

I guess NFS is the a big one still on the todo list.

The point is that there are only a handful of things that we know
about that we still need to convert that make a difference in
practice.

In addition the semantics of the pid namespace make a very big
difference in understanding how we need to group processes. Having
code people can look at an play with makes the subject a lot more
approachable.

Most of the remaining conversions do not actually make sense without
the pid namespace so we have work to do there.

Largely I am trying to structure this in a fashion that is accessible
to more people, which means more people can work on it together.

I think it would be reasonable to not merge the patch that enables
clone/unshare support upstream until we have everything else finished.

I have no intention of declaring a pid namespace done or complete
until it is but getting as close as we can get would be a real
advantage.

>> - When we do the rename can we please rename it task_proxy and have
>>   the functions follow that naming. The resource limiting conversation
>>   seems to be going in that direction, and it more general then what we
>>   are using now.
>
> hmm, nsproxy was unusual but kind of understandable,
> task_proxy sounds just weird to me, I'd definitely
> prefer nsproxy over task_proxy, but I'm open for
> more 'space' related names too, like spaces or
> space_proxy or space_group ...
>

Well it is a proxy for task_struct and task_struct_proxy is just
long winded. Calling it task_proxy makes sticking the pointers
to other subsystems per task data more reasonable.

Eric

_____

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

Subject: Re: Pid namespace patchsets review
Posted by serue on Sun, 11 Mar 2007 13:36:43 GMT
View Forum Message <> Reply to Message

Quoting Eric W. Biederman (ebiederm@xmission.com):
>
> It is good to see these patches are starting to come together.
>
> Be patient a good review is going to take me a little bit.
>
> A couple of immediate things I see that would be nice to address before
> we aim at merging these patches upstream.
>
> - Since there are known cases that we still need to convert to use struct
> pid can we disable the clone/unshare unless we have the CONFIG_EXPERIMENTAL
> flag set. And a comment in Kconfig saying we are almost but not quite
> there yet. With that in place I would have no problems with the idea
> of merging all of the bits needed to have multiple pid namespaces before
> we finish making the code pid namespace safe.
>
> - When we do the rename can we please rename it task_proxy and have the functions
> follow that naming. The resource limiting conversation seems to be going in
> that direction, and it more general then what we are using now.

If we're going to put the resource stuff in, then I agree let's rename.
If we stick to this being a namespace proxy (my preference) then calling
it nsproxy is more accurate.

(I can't keep up with that thread so maybe that's been decided by now :)

-serge

_____

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

Subject: Re: Pid namespace patchsets review
Posted by Herbert Poetzl on Sun, 11 Mar 2007 14:26:30 GMT

On Sat, Mar 10, 2007 at 06:57:13PM -0700, Eric W. Biederman wrote:
> Herbert Poetzl <herbert@13thfloor.at> writes:
>
> > > IMHO not the best idea, mainly because both OpenVZ
> > > and Linux-VServer will end up either duplicating
> > > the pid code or using the incomplete (broken) version
> > > which probably gives the pid space a bad start ...
> > >
> > > I'd prefer to focus on fixing up the existing pid
> > > issues (conversion) first, then hitting it with a
> > > hopefully working pid namespace ...
> > >
> > > YMMV
>
> Right now if we discount the kernel_thread to kthread conversion
> we are probably 98% done with all of the conversions that make sense
> without a pid namespace.
>
> I guess NFS is the a big one still on the todo list.
>
> The point is that there are only a handful of things that we know
> about that we still need to convert that make a difference in
> practice.
>
> In addition the semantics of the pid namespace make a very big
> difference in understanding how we need to group processes. Having
> code people can look at an play with makes the subject a lot more
> approachable.
>
> Most of the remaining conversions do not actually make sense without
> the pid namespace so we have work to do there.
>
> Largely I am trying to structure this in a fashion that is accessible
> to more people, which means more people can work on it together.
>
>
> I think it would be reasonable to not merge the patch that enables
> clone/unshare support upstream until we have everything else finished.
>
> I have no intention of declaring a pid namespace done or complete
> until it is but getting as close as we can get would be a real
> advantage.

sure, I'm perfectly fine with keeping all that stuff
in -mm and test the hell out of it ... no problem
to make a new Linux-VServer branch based on -mm which
provides folks interested in testing to exercise the

pid namespace stuff ...

just in mainline, it would be a bad idea (IMHO)

> >> - When we do the rename can we please rename it task_proxy and
> >> have the functions follow that naming. The resource limiting
> >> conversation seems to be going in that direction, and it more
> >> general then what we are using now.
> >
> > hmm, nsproxy was unusual but kind of understandable,
> > task_proxy sounds just weird to me, I'd definitely
> > prefer nsproxy over task_proxy, but I'm open for
> > more 'space' related names too, like spaces or
> > space_proxy or space_group ...
>
> Well it is a proxy for task_struct and task_struct_proxy is just
> long winded. Calling it task_proxy makes sticking the pointers
> to other subsystems per task data more reasonable.

interesting view, for me it always was a proxy for
the (name)spaces, as for me, the direction always
is task -> proxy -> space, not the other way round

but I'm not going to insist in naming that differently
(i.e. if the majority finds that naming intuitive,
I'm fine getting myself used to it)

best,
Herbert

> Eric

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

Subject: Re: Pid namespace patchsets review
Posted by ebiederm on Sun, 11 Mar 2007 17:45:59 GMT
View Forum Message <> Reply to Message

"Serge E. Hallyn" <serue@us.ibm.com> writes:

> If we're going to put the resource stuff in, then I agree let's rename.
> If we stick to this being a namespace proxy (my preference) then calling
> it nsproxy is more accurate.

Sounds like a reasonable criteria.

> (I can't keep up with that thread so maybe that's been decided by now :)

I got a little overwhelmed as well. That mess needs sorting out
though so I'm going to wade in as soon as have caught up on the bits
with more agreement.

Eric

_____

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