Subject: Re: containers development plans (July 10 version)
Posted by serge on Thu, 12 Jul 2007 18:45:00 GMT
View Forum Message <> Reply to Message

Quoting Kirill Korotaev (dev@sw.ru):
> Serge E. Hallyn wrote:
> > (If you missed earlier parts of this thread, you can catch earlier parts of
> > this thread starting at
> > https://lists.linux-foundation.org/pipermail/containers/2007 -July/005860.html)
> >
> > Thanks for all the recent feedback. I particularly added a lot from Paul
> > Menage and Cedric.
> >
> > We are trying to create a roadmap for the next year of
> > 'container' development, to be reported to the upcoming kernel
> > summit. Containers here is a bit of an ambiguous term, so we are
> > taking it to mean all of:
> >
> > 1. namespaces
> > kernel resource namespaces to support resource isolation
> > and virtualization for virtual servers and application
> > checkpoint/restart.
> > 2. task containers framework
> > the task containers (or, as Paul Jackson suggests, resource
> > containers) framework by Paul Menage which especially
> > provides a framework for subsystems which perform resource
> > accounting and limits.
> > 3. checkpoint/restart
> >
> > A (still under construction) list of features we expect to be worked on
> > next year looks like this:
> >
> > 1. completion of ongoing namespaces
> > pid namespace
> > merge two patchsets
> sukadev@ and Pavel already agreed and will resend it soon
> > clone_with_pid()
> > kthread cleanup
> > especially nfs
> > autofs
> > af_unix credentials (stores pid_t?)
> > net namespace
> > ro bind mounts
>
> IMHO ro bind mounts are not related to namespaces anyhow, but ok if you guys want to
mention it.

Hmm, yes it's more for the "userspace containers" - meaning the

userspace usage of namespaces. But I'm not sure it's worth breaking
that out.

> > sysvipc
> > "set identifier" syscall
>
> the last one is related to checkpointing, so plz move it from here...

It started under checkpointing, but I'll move it back :)

> > 2. continuation with new namespaces
> > devpts, console, and ttydrivers
> > user
> > time
> > namespace management tools
> > namespace entering (using one of:)
> > bind_ns()
> > ns container subsystem
> > (vs refuse this functionality)
> > multiple /sys mounts
> > break /sys into smaller chunks?
> > shadow dirs vs namespaces
> > multiple proc mounts
> > likely need to extend on the work done for pid namespaces
> > i.e. other /proc files will need some care
>
> different statistics virtualization here in /proc for top and other tools
>
> > 3. any additional work needed for virtual servers?
> > i.e. in-kernel keyring usage for cross-usernamespace permissions, etc
> > nfs and rpc updates needed?
> > general security fixes
>
> what is meant by "general security fixes"?

I think it means "we haven't thought it through enough" :)

For instance, something needs to be done to be able to hand
partial capabilities to admins in a container/virtual server. We've
talked about doing this using the in-kernel keyring, but we are far from
consensus or patches, and this will have to be solved.

> what I see additionaly:
> - device access controls (e.g. root in container should not have access to /dev/sda by default)

Yes, that kind of falls under the above, but I'll add it separately.

> - filesystems access controls

ditto.

> > 4. task containers functionality
> > base features
> > virtualized continerfs mounts
> > to support vserver mgmnt of sub-containers
> > locking cleanup
> > control file API simplification
> > control file prefixing with subsystem name
> > specific containers
> > usespace RBCE to provide controls for
> > users
> > groups
> > pgrp
> > executable
> > split cpusets into
> > cpuset
> > memset
> > network
> > connect/bind/accept controller using iptables
> > network flow id control
> > userspace per-container OOM handler
>
> I don't see much about resource management here at all.
> We need resource controls for a lot of stuff like
> - RSS
> - kernel memory and different parameters like number of tasks
> - disk quota
> - disk I/O
> - CPU fairness
> - CPU limiting
> - container aware OOM
>
> imho it is all related and should be discussed.
>
> > 5. checkpoint/restart
> > memory c/r
> > (there are a few designs and prototypes)
> > (though this may be ironed out by then)
> > per-container swapfile?
> > overall checkpoint strategy (one of:)
> > in-kernel
> > userspace-driven
> > hybrid
> > overall restart strategy
> > use freezer API
> > use suspend-to-disk?

> >
> > In the list of stakeholders, I try to guess based on past comments and
> > contributions what *general* area they are most likely to contribute in.
> > I may try to narrow those down later, but am just trying to get something
> > out the door right now before my next computer breaks.
> >
> > Stakeholders:
> > Eric Biederman
> > everything
> > google
> > containers
> > ibm
> > everything
> > kerlabs
> > checkpoint/restart
> > openvz
> > everything
> > osdl (Masahiko Takahashi?)
> > checkpoint/restart
> > Linux-VServer
> > namespaces+containers
> > zap project
> > checkpoint/restart
> > planetlab
> > everything
> > hp
> > ?
> > XtreemOS
> > checkpoint/restart
> >
> > Is anyone else still missing from the list?
> >
> > thanks,
> > -serge
> >

thanks Kirill,

-serge
