
Subject: Re: [RFC] Container mini-summit agenda for Sept 3, 2007

Posted by [Oren Laadan](#) on Fri, 31 Aug 2007 03:26:22 GMT

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

Cedric Le Goater wrote:

> Hello All,
>
> Some of us will meet next week for the first mini-summit on containers.
> Many thanks to Alasdair Kergon and LCE for the help they provided in
> making this mini-summit happen !
>
> It will be held on Monday the 3rd of September from 9:00 to 12:45 at LCE
> in room D. We also might get a phone line for external participants and,
> if not, we should be able to set up a skype phone.
>
> Here's a first try for the Agenda.
>
> Global items
>
> [let's try to defer discussion after presentation]
>
> * Pavel Emelianov status update
> * Serge E. Hallyn Container Roadmap including
> . task containers (Paul Menage)
> . resource management (Srivatsa Vaddagiri)
>
> Special items
>
> [brainstorm sessions which we would like to focus on]
>
> * building the global container object ('a la' openvz or vserver)
> * container user space tools
> * container checkpoint/restart

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?

sysvipc

"set identifier" syscall

pid namespace

clone_with_pid()

There are other identifiers - pseudo terminals, message queues (mq) (if you insist on supporting these ...). In general, we need a way to specify the virtual id of a resource that is created. I suggest that this should be part of an interface between c/r and containers (see below)

live migration

aka pre-copy (which can be used for live migration but also to reduce the downtime due to a checkpoint).

how about adding incremental checkpoint to the list ?

I think that it is also important to discuss an interface between c/r and containers, each of which stands on it own. For instance, how to request a specific virtual id (during restart), define required notifiers (to set/unset c/r related data on/off a task), control c/r-related setting of container (e.g. frozen, restarting) that may affect behavior, such as signal handling, and so forth. Also, such an interface can allow existing c/r implementations to work with different virtualization implementations as they become available.

Many of these were discussed in a recent Zap paper present in USENIX:
http://www.ncl.cs.columbia.edu/publications/usenix2007_fordist.pdf
The paper describes important design choices in Zap (but I'm biased ...).
I think it may serve as an appetizer for the discussion :P

Oren.

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>
