

---

Subject: containers development plans (July 10 version)

Posted by [serge](#) on Tue, 10 Jul 2007 21:39:43 GMT

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

---

(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
    - clone\_with\_pid()
    - kthread cleanup
      - especially nfs
      - autofs
    - af\_unix credentials (stores pid\_t?)
  - net namespace
  - ro bind mounts
  - sysvipc
    - "set identifier" syscall
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
- 3. any additional work needed for virtual servers?
  - i.e. in-kernel keyring usage for cross-namespace permissions, etc
  - nfs and rpc updates needed?
  - general security fixes
- 4. task containers functionality
  - base features
    - virtualized containerfs mounts
      - to support vserver mgmnt of sub-containers
    - locking cleanup
    - control file API simplification
    - control file prefixing with subsystem name
  - specific containers
    - userspace 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
- 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

---